

Administración de sistemas Linux



De todos los sistemas Unix, Linux es una plataforma servidora excelente, un buen sistema de escritorio y el centro en torno al cual gira una gran parte de la innovación del mundo informático actual. Linux es probablemente el que más ámbitos abarca de todos los sistemas operativos, desde sistemas pequeños como un teléfono móvil hasta clústeres de computadores más grandes que un edificio. Está presente en los campos de las telecomunicaciones, sistemas embebidos, satélites, equipamiento médico, sistemas militares y gráficos por computador e informática de escritorio.

Administración de sistemas Linux ofrece numerosos consejos para gestionar un amplio rango de sistemas y servidores. Este libro resume los pasos para implantar desde hubs SOHO, servidores web y servidores LAN, hasta clústeres de carga balanceada y servidores virtuales. También le ayudará a conocer las herramientas necesarias para administrar y mantener de forma eficaz estos entornos de trabajo.

Con este libro aprenderá a:

- Instalar, configurar, mantener y resolver problemas en un servidor usando BIND.
- Configurar un servidor de correo Postfix con autentificación SASL, un servidor POP y un servidor IMAP.
- Gestionar usuarios y configurar elementos de red comunes, tales como DCHO y software para pasarelas en redes de área local (LAN).
- Definir Xen, VMware en un equipo Linux y luego añadir sistemas operativos invitados.
- Instalar y configurar Apache, PHP y MySQL en un servidor web desde cero.
- Hacer copias de seguridad y restaurar datos con rsync, tar, cdrecord, Amanda y herramientas MySQL.

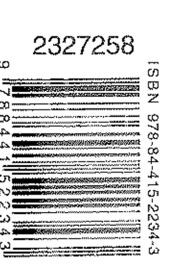
Tom Adelstein comenzó su carrera en el mundo de las inversiones bancarias, donde sus conocimientos técnicos ayudaron a algunas empresas de servicios financieros a convertirse en líderes de su sector. Ahora es administrador de sistemas y escritor técnico.

Bill Lubanovic comenzó desarrollando software para Unix en la década de 1970, interfaces gráficas de usuario en la década de 1980 y para la Web en la década de 1990. Actualmente trabaja en el área de visualización Web para una compañía de energía eólica.





2327258



ADMINISTRACIÓN DE SISTEMAS LINUX

Tom Adelstein
Bill Lubanovic

ADMINISTRACIÓN DE SISTEMAS LINUX

ANAYA MULTIMEDIA
O'REILLY

Tom Adelstein
Bill Lubanovic

O'REILLY

Título de la obra original:
Linux System Administration

Responsable editorial:
Víctor Manuel Ruiz Calderón
Alicia Cózar Concejil

Traducción:
Jorge Martínez Gil

Realización de cubierta:
Sandra Cordova Yusta



Administración de sistemas Linux

Tom Adelstein
Bill Lubanovic



Juan Ignacio Luca de Tena, 15. 28027 Madrid.
Depósito legal: M-31.807-2007
ISBN: 978-84-415-2234-3
Printed in Spain.
Imprime: Artes Gráficas Guemo, S.L.
Febrero, 32. 28022 Madrid.

Agradecimientos

Este libro no existiría sin la contribución de muchas personas. Aunque es imposible incluirlas aquí a todas. En primer lugar, queremos dar las gracias a Andy Oram cuyos esfuerzos de edición, redacción y gestión son destacables. Aparte de su trabajo como editor, Andy ha contribuido materialmente al contenido de este libro. Andy ha ejercido como gestor del proyecto y ha demostrado paciencia y disciplina.

Tampoco podemos olvidar las contribuciones de Falko Timme, Phil Howard y Herschel Cohen. Falki nos ha prestado su tiempo y experiencia en los capítulos 2 y 4. Phil ha escrito gran parte del capítulo 11 y nos ha proporcionado el framework del capítulo 10 y el apéndice de los scripts. Herschel ha escrito varias secciones de capítulos, entre los que están el capítulo 8 y 10, y ha contribuido con su experiencia en el capítulo 6. Los tres, además, han revisado otras partes del libro.

Muchas gracias también a los expertos técnicos, que han pasado innumerables horas revisando, probando y haciendo sugerencias sobre el trabajo: Markus Amersdorfer, Keith Burgess, Robert Day, Ammar Ibrahim, and Yaman Saqqa.

Y un agradecimiento especial a Yvonne Adelstein y Mary Lubanovic, nuestras esposas, que han demostrado una gran paciencia. No habríamos podido realizar este trabajo sin su apoyo.

Sobre los autores

Tom Adelstein comenzó su carrera en el mundo de las inversiones bancarias, donde sus conocimientos técnicos ayudaron a algunas empresas de servicios financieros a convertirse en líderes de su sector. Ahora es administrador de sistemas y escritor técnico.

Bill Lubanovic comenzó desarrollando software para Unix en la década de 1970, interfaces gráficas de usuario en la década de 1980 y para la Web en la década de 1990. Ahora trabaja en el área de visualización Web para una compañía de energía eólica.

Contenido



Introducción



Cuando Bill Lubanovic y yo estábamos dando los retoques finales a este libro escuché, sin quererlo, una conversación entre dos compañeros de trabajo en nuestro laboratorio Cisco acerca de Linux. Uno de los dos era experto en redes y hacía una puntualización muy interesante. Decía que a pesar de todos sus conocimientos, se sentía un profesional incompetente porque nunca había aprendido Linux. Un poco después, el otro hombre se giró y me miró. Sonreí y continué trabajando.

Por la noche, nuestro director de Tecnologías de la Información me hizo un comentario improvisado y nada usual durante una conferencia. Me dijo que quería aprender Apache, y cuando le pregunté por qué, me respondió: "Simplemente quiero aprender", y la cosa quedó ahí.

Después en la conferencia, nuestro director pidió al grupo una solución para la gestión de parches, explicando y haciendo uso de rsync como ejemplo. Dijo que quería algo similar, mientras entraba en detalles acerca de una herramienta para la gestión de parches de manera incremental y acumulativa.

En ambos casos, y en muchos otros, deseé tener este libro acabado para recomendárselo a toda esa gente que aunque poseía experiencia y habilidades, quería aprender a administrar Linux. Quizás usted tenga experiencias similares y le hubiese gustado tener un libro como este a mano en aquellos momentos. Sospecho que conversaciones parecidas a las que acabo de contar ocurren muy a menudo en muchos lugares, incluso diariamente.

Cuando Andy Oram y yo comenzamos a debatir acerca de un libro para la administración de sistemas Linux, teníamos una idea ligeramente diferente de lo que queríamos hacer. Andy hablaba de un libro en el que cada capítulo mostrará a los usuarios los pasos para construir y desplegar servidores sin entrar en deta-

lles. Él proponía debatir primero acerca de los capítulos y luego acerca de los pasos técnicos.

Después, yo propuse que hiciéramos de cada capítulo un módulo y así permitir al lector completar los módulos que quisiera o que necesitase. A medida que el libro evolucionaba, sentíamos que se estaba cumpliendo este objetivo. No es necesario que lea este libro de principio a fin para convertirse en un administrador de sistemas Linux. Simplemente, comience por donde le interese.

Cuando yo empecé a manejar Linux, la comunidad estaba compuesta en su mayor parte por programadores y aficionados. No creo recordar ninguna lista de discusión que se centrará en aplicaciones comerciales o de escritorio. Accedíamos a Internet mediante un demonio de inicio. No teníamos conexiones ni navegadores Web como los que están disponibles hoy en día.

La gran mayoría de la gente que yo conocía o eran administradores de su propio sistema o estaban aprendiendo. Me estoy refiriendo a una época en la que los usuarios de Linux eran unos 30.000 en todo el planeta, todavía me sorprendo con la cantidad de gente que usa Linux a día de hoy sin tener la menor idea de cómo escribir un archivo de configuración. Los foros de Linux se llenan de personas que preguntan qué tienen que hacer para conseguir que CUPS o Samba funcionen.

En las listas de correo, la gente mantiene debates acerca de detalles técnicos de proyectos como Postfix, JBoss y Monit. Muchas personas aún sienten curiosidad por aprender las grandes posibilidades de Linux como plataforma de aplicaciones. Si usted ya usa Linux a nivel de usuario, y quiere ir un paso más allá, es decir, quiere convertirse en administrador, este libro le ayudará en la transición, puesto que hemos escrito este libro pensando en usted.

Cómo se organiza el libro

- **Capítulo 1. Requisitos para un administrador de sistemas Linux:** Repasa los objetivos del libro y qué se conseguirá al leerlo.
- **Capítulo 2. Configurando un servidor Linux multifunción:** Le inicia en el manejo de servidores sencillos para Internet.
- **Capítulo 3. El sistema de nombres de dominio:** Muestra los aspectos básicos para configurar servidores DNS primarios y secundarios.
- **Capítulo 4. Un entorno inicial listo para Internet:** Usa el software de configuración ISPConfig para introducirle en un conjunto de servicios con los que podrá practicar mientras lee el resto del libro.
- **Capítulo 5. Correo:** Configura un servidor de correo Postfix con autentificación SASL, un servidor POP y un servidor IMAP.

- **Capítulo 6. Administrando Apache:** Proporciona una visión rápida de la popular combinación Apache, MySQL y PHP (a la que junto con Linux se le llama servidor LAMP), incluyendo autentificación SSL.
- **Capítulos 7. Clusters de carga balanceada:** Extiende el capítulo previo de configuración de Apache con el servidor de IP Virtual y ldirectord para ofrecer gran capacidad.
- **Capítulo 8. Servicios de red de área local:** Muestra cómo gestionar usuarios y configurar elementos de red comunes, tales como DCHO y software para pasarelas en redes de área local (LAN).
- **Capítulo 9. Virtualización de la empresa moderna:** Muestra cómo definir Xen, VMware en un equipo Linux y luego añadir sistemas operativos invitados.
- **Capítulo 10. Scripting:** Muestra algunas técnicas básicas para escribir script potentes y robustos que pueden ahorrar mucho tiempo de administración.
- **Capítulo 11. Haciendo copia de seguridad de los datos:** Presenta un amplio rango de técnicas para realizar esta función crucial, desde el rsync básico y el tar hasta el potente sistema Amanda.
- **Apéndice. Bash scripts de ejemplo:** Contiene unos cuantos shell scripts que nos han sido útiles para administrar sistemas y también proporciona consejos a tener en cuenta a la hora de escribir tus propios scripts.

Convenciones usadas en este libro

Para ayudarle a sacar el mayor partido al texto y saber dónde se encuentra en cada momento, a lo largo del libro utilizamos distintas convenciones:

- Las combinaciones de teclas se muestran en negrita, por ejemplo **Control-A**. Los botones de las distintas aplicaciones también se muestran en negrita.
- Los nombres de archivo, URL y código incluido en texto se muestran en un tipo de letra `monoespacial`.
- Los menús, submenús, opciones, cuadros de diálogo y demás elementos de la interfaz de las aplicaciones se muestran en un tipo de letra Arial.

Nota: En estos cuadros se incluye información importante directamente relacionada con el texto adjunto. Los trucos, sugerencias y comentarios afines relacionados con el tema analizado se reproducen en este formato.

Capítulo 1



Requisitos para un administrador de sistemas Linux



Nos gusta Linux. De todos los sistemas Unix y parecidos a Unix que hemos usado, muchos ahora olvidados, Linux es nuestro favorito. Es una plataforma servidora excelente, un buen sistema de escritorio y el centro en torno al cual gira mucho de la innovación del mundo informático actual.

Linux es probablemente el que más aspectos abarca de todos los sistemas operativos, desde sistemas pequeños como un teléfono móvil hasta clústeres de computadores más grandes que un edificio. Está presente en los campos de las telecomunicaciones, sistemas embebidos, satélites, equipamiento médico, sistemas militares y gráficos por computador, y por último, pero no por ello menos importante, informática de escritorio.

En un período de tiempo relativamente corto, Linux ha pasado de ser el pasatiempo de un hacker finlandés a un sistema avanzado de nivel empresarial respaldado por gigantes como IBM y Oracle. La base de usuarios ha crecido considerablemente desde las 30.000 personas que había en 1995 hasta los cientos de millones que hay a día de hoy. Durante el boom de Internet en la década de 1990, muchos administradores de Unix se sorprendieron gratamente al descubrir que un Linux sobre un PC podía hacer las mismas tareas que carísimas estaciones y servidores UNIX. Muchos administradores de Windows y de Novell vieron que Linux podía manejar DNS, correo electrónico y servicios de archivos más eficientemente y con menos soporte humano que sus diferentes plataformas. El crecimiento de Internet, y especialmente la Web, sirvió de combustible para la rápida expansión del uso de los servidores Linux y la necesidad de personal para gestionarlos.

Este libro es para administradores de sistemas Linux. No obstante, puede que sea un veterano de Unix, un bravo MCSE o un estoico administrador de

mainframes. Pero está explorando un nuevo territorio y necesita mapa y compás. Algunos aspectos le sonarán familiares, pero otros serán una tierra sin explorar. Este libro cubre muchos temas que se acaban de añadir a la tendencia actual, por ejemplo los clústeres de carga balanceada y la virtualización.

El éxito de Internet y del software de código abierto está cambiando los negocios. Google, Amazon, eBay y otros han levantado granjas de servidores con hardware manejable y relativamente pocos administradores en comparación con las instalaciones de PC y mainframes tradicionales.

Los conocimientos necesarios para desarrollar y mantener tales sistemas distribuidos y las aplicaciones no se enseñan en los colegios, sino que se aprenden de la experiencia, unas veces más amarga y otras más dulce.

Nota: Mientras escribíamos este libro hemos estado de forma constante probando las últimas distribuciones y herramientas, y mantendremos nuestros experimentos hasta después de que el libro esté terminado. Invitamos a los lectores a que visiten el sitio Web que hemos levantado por el libro, `http://www.centralsoft.org`, donde publicaremos actualizaciones de los ejemplos, enlaces a nuevas y útiles herramientas que vayamos descubriendo y otros consejos.

Sobre este libro

Los libros de administración de sistemas suelen ser fácilmente predecibles. Enseñan cómo gestionar usuarios, sistemas de archivos, dispositivos, procesos, impresoras, redes, etc. Pero no indican qué tiene que hacer cuando surge un problema. Si su sitio Web se convierte en popular, tendrá que aprender rápidamente a usar servidores proxy, diferentes niveles de caché, balanceado de carga, autentificación distribuida y otros detalles complejos. Si añade una base de datos, necesitará saber cómo ampliarla y aprender a evitar los ataques de inyección SQL. De la noche a la mañana, el sitio se ha convertido en una misión crítica, y deberá ser capaz de hacer copias calientes en sistemas 24 x 7.

Si ya ha realizado muchos simulacros de incendios, estará cansado de hacerlo todo de la manera más difícil, pero tendrá que hacer frente a nuevos retos técnicos prácticamente a diario y con la ayuda de muy pocas fuentes de ayuda. La documentación técnica, ya sea de software comercial o software libre, rara vez tiene que ver con la tecnología, y suele pecar de ser demasiado abstracta. Por ejemplo, los servidores de directorios de código abierto se han convertido en algo importante para gestionar ordenadores, usuarios y recursos. Los productos comerciales suelen cumplir los protocolos del RFC original, pero la buena documentación para proyectos abiertos es sorprendentemente escasa.

¿Cómo podemos ayudarle?

Las personas que trabajan en Linux resuelven problemas. Un usuario medio de Linux puede levantar un pequeño servidor, obtener una conexión con una IP estática en su casa, registrar un nombre de dominio y poner en marcha el servidor en Internet. Si es de los que están en esta categoría, puede leer los otros temas del libro y ampliar sus posibilidades profesionales. A algunos, todo esto les parecerá como escalar una montaña de 10.000 metros. Si no es uno de ellos, comience por cualquier parte. Como dice el refrán, come al elefante de una vez y luego saboréalo.

Quizás tenga certificaciones de otros sistemas operativos distintos a Linux. Aunque ya sabrá aplicar parches y corregir fallos, aquí aprenderá a desplegar el servidor Apache, a manejar su propio DNS o a cambiar Exchange por Zimbra.

Si solo quiere aprender o tiene la obligación de aprender, necesitará ayuda para escalar la curva de aprendizaje de Linux. Para eso es exactamente para lo que estamos: para ayudarle a explorar el sistema Linux sin tener que pasar por experiencias traumáticas.

¿Por dónde empezar?

Este libro resume los pasos que tiene que seguir para desplegar servidores autónomos. Si necesita levantar un servidor de correo, crear un servidor Web con capacidad para blogs o configurar una pasarela para su LAN, puede dirigirse a la mitad del libro. No tiene que leer "Administración de sistemas Linux" desde el principio hasta el final.

Comenzaremos explicando, paso por paso, la manera levantar un servidor Linux en el capítulo siguiente. Puede elegir el camino que le convenga, desde crear un cluster para servicios Web o reforzar los servidores gracias a la virtualización, hasta configurar un servidor para una red de área local.

Hacer funcionar un sistema operativo moderno es muy barato. Puede configurar un sofisticado centro de aprendizaje sobre hardware que en muchos sitios consideran obsoleto y lo regalan. Nosotros comenzamos con una caja que contenía una CPU de Intel dos generaciones más antigua que los modelos actuales, le pusimos una versión antigua de discos duro y memoria, y una versión sin extras y gratuita de Linux.

¿Necesita un libro?

Los libros técnicos han adquirido popularidad a medida que Internet ha madurado. Para conseguir un libro de éxito hoy en día, el autor tiene que proporcio-

nar un valor añadido al lector. Una historia interesante sobre uno de los primeros sitios de comercio electrónico ayuda a explicar qué debería ofrecer un libro. Una compañía que fabricaba tartas de queso puso un anuncio en los primeros días de la Web. Según la historia, pasaron varios meses sin que la compañía recibiera un pedido. En un movimiento nada común, el presidente de la compañía publicó la receta secreta de la tarta de queso. Al cabo de unas horas, su línea de teléfono estaba colapsada. La gente empezó a pedir muchos pasteles de queso. Se dieron cuenta de que se necesitaba un esfuerzo considerable para hacer sus propios pasteles y por tanto vieron el valor de comprárselos a la compañía.

Muchos de los ingredientes de este libro han sido recopilados de Internet, de listas de correos, foros, grupos de discusión, mientras que otros han sido extraídos de libros, revistas y la experiencia de amigos. Nosotros solucionamos muchos problemas cuyas soluciones estaban completamente indocumentadas antes de aparecer este libro, y ahora se las ofrecemos.

Muchos proyectos excelentes tienen una documentación inadecuada. Los desarrolladores han trabajado mucho para ofrecer un excelente software libre, pero no una documentación adecuada para el código por varias razones: falta de tiempo, falta de recursos, falta de interés, la barrera del lenguaje, etc.

Junto con nuestros lectores, editores y revisores, esperamos que esta tendencia disminuya, al menos en esta pequeña parcela del mundo de la informática.

¿Quién le necesita?

Hace unos cuantos años, la mayoría de los administradores de sistemas Linux decían que ellos no había escogido esta carrera, sino que Linux les había escogido a ellos. En aquellos tiempos, Linux era como un Unix adolescente. La mayoría de los administradores de sistemas Linux aprendían el funcionamiento en una estación de trabajo y en pequeñas redes. Linux heredó algunos servidores de Unix (BIND, Sendmail, Apache), pero también software de oficina y algunas aplicaciones. Hoy en día, la administración de sistemas Linux involucra a miles de paquetes y la interoperabilidad con otros sistemas operativos.

¿Quién necesita administradores Linux? El centro para ciencias de la computación de la *NASA (NCSS) y el Centro para vuelos espaciales Goddard*. Los clústeres para computación de alto rendimiento basados en Linux se diseñan para mejorar el tiempo de respuesta de aplicaciones, que van desde el estudio del tiempo y del cambio climático hasta la simulación de fenómenos astrofísicos. Linux permite a la arquitectura de la NCSS soportar hasta 4 billones de operaciones en coma flotante por segundo (TFLOPS) en su configuración de alto rendimiento. Linux es el sistema operativo que más supercomputadores hace funcionar en el mundo. De hecho, a día de hoy da soporte a un 75 por 100 de los 500 supercomputadores que hay en el planeta. Según el *Lawrence Livermore National Laboratory* de Livermore (California), Linux se ejecuta en 10 de sus sistemas más robustos, todos incluidos entre la lista de los 500. Entre estos sistemas se incluye el BlueGene/L, el supercomputador más potente, y el Thunder, que actualmente ocupa el puesto diecinueve (`http://www.top500.org/list/2006/11/100`).

Ayuda demandada

Los administradores Linux están muy solicitados. Para que se haga una idea de que se espera de ellos, hemos recopilado una pequeña selección de algunas de las miles de tareas que un administrador Linux puede desempeñar según un sitio Web de una agencia nacional de empleo. He aquí algunas de las responsabilidades:

- Administrar y gestionar un gran entorno de servidor Linux, con énfasis en el rendimiento, la monitorización, la personalización y la gestión.
- Supervisar el diseño físico de las bases de datos, su administración y su documentación.
- Ofrecer soluciones a los problemas que puedan surgir en la red, dar soporte a la ampliación de servicios y la monitorización proactiva de sistemas críticos.
- Ofrecer consejos y aportar soluciones tecnológicas a la organización; entrenar y tutorizar a los administradores noveles.
- Ofrecer soporte técnico diario y resolver consultas relativas al hardware y al sistema operativo; administrar la infraestructura del servidor Linux para mantener la estabilidad, así como maximizar la eficiencia del entorno.
- Instalar, configurar y solucionar los fallos relativos al hardware, periféricos y equipamiento necesario para mantener la integridad del sistema; ofrecer soporte para ampliaciones.
- Ofrecer soporte efectivo de primer/segundo nivel para entornos Linux a través de sus servidores.
- Gestionar todos los aspectos relativos a la integridad del entorno, incluyendo la seguridad, la monitorización (de la capacidad y del rendimiento), el control de cambios y la gestión del software.
- Interactuar con otros grupos de soporte interno como el de Control de Cambios, Desarrollo de Aplicaciones, Ingeniería, Administradores de Bases de Datos, Servicios Web, Almacenamiento, Seguridad, Operaciones y Centros de Órdenes.
- Administrar servicios de infraestructura como: DNS, NIS, LDAP, FTP, SMTP, Postfix/Sendmail, NFS, SAMBA, y servidores de aplicaciones y de bases de datos, con énfasis en la automatización y en la monitorización.

Linux es ahora una plataforma estándar y el talento no abunda. Si quieres aprender Linux podrás mejorar tu salario, como así lo evidencia la creciente demanda de trabajadores con conocimientos de administración Linux.

Analizando los conjuntos de habilidades

Pregunte a distintos gestores de sistemas de información sobre el papel que debe jugar el administrador del sistema y obtendrá una gran variedad de respuestas. El mercado se ha visto sorprendido por el hecho de que la mayoría de estos gestores carecen de conocimientos acerca de Linux. Ellos no saben qué es lo que deberían saber los profesionales de Linux, y los profesionales de Linux, rara vez entienden a estos gestores.

Muchos gestores de sistemas de información que entienden Unix quieren adaptar a los administradores de Linux a los estándares de Unix. Esto no suele funcionar. Aunque los administradores Unix crean que la migración a Linux es fácil, siempre suelen descubrir que no es así. Los administradores Linux tienen menos problemas para adaptarse a Unix que al contrario. Una explicación es que los administradores Linux tiene un conocimiento mucho más amplio de sus sistemas, debido a la naturaleza del software del código abierto.

Las tareas de los administradores de sistemas muy a menudo involucran a Internet. La mayoría de las transacciones están relacionadas con el correo y la gestión de sitios Web, además de las telecomunicaciones y la movilidad. El correo electrónico representa el 70 por 100 de todo el tráfico de Internet. Hoy en día, las aplicaciones de banda ancha como la Voz sobre IP (VoIP) y otras formas de comunicación, incluyendo la mensajería instantánea, han incrementado el tráfico en detrimento del correo electrónico. Pero a pesar de los protocolos y las aplicaciones multimedia, Internet sigue siendo el dominio principal de Linux.

Continuemos analizando las responsabilidades laborales descritas en la sección previa. La última tarea ("Administrar servicios de infraestructura") puede darle una idea del conjunto de habilidades necesarias en Linux. Las empresas quieren administradores de sistemas que puedan manejar "infraestructuras de servicios". Fíjese en las tecnologías de Internet involucradas. De la lista de componentes Linux que es necesario saber, la mayoría de las tareas involucran DNS, LDAP, FTP, SMTP y Postfix/Sendmail. Cubriremos la mayor parte de estos componentes en próximos capítulos.

Las otras descripciones de trabajos se ajustan más a la categoría de necesidades internas de la empresa. Entre ellas se incluyen el soporte para el diseño de servicios escalables, soporte técnico y consultor para el entorno del hardware y del sistema operativo. La mayoría de los administradores de sistemas Linux deberían poseer habilidades para ofrecer estos servicios, pero éstos están fueran del alcance del libro porque no son de naturaleza técnica.

El resto de las responsabilidades se encuadran dentro de la categoría de "habilidades ligeras". En el pasado, nadie esperaba que un administrador de sistemas aprendiera a funcionar como enlace con otros grupos internos de soporte como el de Desarrollo de Aplicaciones, Ingeniería, Administradores de Bases de datos o Servicios Web. Sin embargo, un administrador de sistemas no es sólo un técnico con conocimientos de algunos sistemas misteriosos, sino el miembro de un grupo capaz de tomar decisiones.

Uno normalmente consigue unas habilidades básicas y especializaciones después de estudiar los aspectos básicos. Quizás cubramos algunas de esas habilidades en este libro, pero creemos que ese no es el objetivo. Otros libros de O'Reilly y el tiempo le ayudarán a adquirir esas habilidades. Por ahora, le introduciremos en el área de la administración de sistema, puesto que es la que mayor crecimiento experimenta y donde no parecer existir una adecuada documentación. Al contrario que en otras disciplinas de informática y de ingeniería, pocos centros ofrecen cursos de Administración Linux, dejando incompletos ciertos programas de grado. Por lo que si usted quiere aprender administración de sistemas Linux, tiene que buscar materiales y cursos fuera de la universidad. Pero una gran parte del material existente no incluye lo que los expertos en Linux consideran los asuntos más importantes.

La mayoría de los administradores Linux son autodidactas, es decir, han aprendido por sí solos. Sin embargo, cuando estos administradores autodidactas han conseguido un trabajo, normalmente no han podido desarrollar su labor correctamente, porque no sabían todo lo que tenían que hacer. Esta es un área a la que Administración de sistemas Linux puede contribuir, ayudándole a mejorar su productividad a la hora de acometer un amplio rango de tareas de manera rápida y eficiente.

Qué deberían saber los gestores de sistemas sobre Linux

Una de las primeras cosas que los gestores de tecnologías de la información deberían saber es que Linux no es Unix. Aunque Linux puede ejecutar la gran mayoría de los programas Unix, también puede ejecutar un amplio rango de aplicaciones sobre redes públicas y privadas.

Los administradores de Linux pueden configurar las distribuciones escogiendo entre un gran número de componentes que hacen trabajos similares. Por ejemplo, en casi todas las distribuciones Unix, Sendmail es la única opción como agente de transferencia de correo (MTA). Pero con Linux, puede elegir entre muchos MTA, dependiendo de si quiere una aplicación para trabajo colaborativo, un *backbone* de correo a gran escala o simplemente una aplicación Web para manejar formularios del tipo "Contacte con nosotros".

Una prueba más de la flexibilidad de Linux es que es el primer sistema operativo que IBM ha escogido para funcionar en todas sus plataformas hardware, desde las series de servidores Intel xSeries o las pSeries y las iSeres, hasta el S/390 y los mainframes zSeries. Si quiere un administrador de Linux y usa grandes sistemas IBM, su candidato debe saber arquitectura de mainframes y estar familiarizado con términos como DASD para almacenamiento en disco duro, IPL para arrancar el sistema, "catalog" para un directorio y "lista de comandos" para una interfaz de comandos. Pero no sea demasiado exigente con ellos. Una vez asistimos a un seminario de dos días con un grupo de administradores de Linux, que se estuvieron en clase un día y luego empezaron a desplegar Linux en ordenadores IBM de la serie zSeries. Si hay algo que la gente de Linux puede ofrecer, es que aprenden rápidamente, se adaptan rápidamente y tienen unos conocimientos que no poseen otros tecnólogos. Pueden aprender a manejar un sistema Microsoft en menos tiempo que lo que tardaría un MCSE en aprender una simple tarea de Linux.

¿Qué es lo próximo?

Sabemos que no le gusta el aprendizaje a paso lento ni los contenidos densos (de hecho estamos sorprendidos de que haya llegado hasta este punto del capítulo), por lo que empezaremos lo más rápido posible. Queremos proporcionar un servidor de trabajo que pueda realizar muchas de las tareas que necesita aprender y usar. Por este motivo, empezaremos a configurar un servidor listo para Internet en el siguiente capítulo. Aprenderá a desplegar un servidor Web y de correo sin importar para qué sea el servidor (incluso aunque esté destinado al soporte de una LAN) y aquellas herramientas que le serán útiles para comenzar.

El resto del libro amplía algunos de los temas e introduce otros con los que se encontrará casi a diario. Administración de sistemas Linux es una combinación de recetario de cocina y guía de viajes; usted podrá disfrutar de su desayuno mientras aprende. Normalmente, explicaremos los conceptos al principio del capítulo y luego los pasos y las aplicaciones concretas. Si sólo quiere seguir paso por paso las instrucciones, puede hacerlo. Podrá resolver su problema después. Si sigue nuestros pasos, le aseguramos que irá en la dirección correcta. ¡Adelante!

Capítulo 2

Configurando un servidor Linux multifunción



Hay una diferencia real entre leer algo y hacer algo. Por ello muchas escuelas disponen de laboratorios para impartir las clases. Si usted quiere aprender administración de sistemas Linux, necesita un servidor. Así es que, la primera tarea de este libro consiste en desplegar un entorno básico. Una vez desplegado, podrá disfrutar de una buena base para practicar y aprender Linux.

El sistema operativo Linux se parece al eje de un coche que puede tener una gran variedad de funciones dependiendo del chasis y de otras características. Es como si al añadir servicios como correo electrónico o bases de datos, el sistema fuera un personaje distinto. ¿Usted necesita un servidor Web, una plataforma de desarrollo, una pasarela o un servidor de archivos y de impresión? Sea lo que sea, necesita un núcleo, que es lo que este capítulo describe. Vamos a comenzar con un servidor que funcione en Internet y que pueda albergar sitios Web. ¿Por qué? Porque es posible adaptar un servidor de Internet a muchas tareas adicionales, como la gestión y autentificación de usuarios, la compartición de archivos y de impresoras, el manejo de correo local y el ofrecimiento de acceso remoto. Puede ofrecer facilidades para el alojamiento Web, configurarlo y empezar a ofrecer servicios Web. Incluso puede convertirlo en su propio sitio, si su ISP le facilita una dirección IP estática.

Configurar un servidor para Internet cambiará su perspectiva de la informática. Desplegar una red de área extensa (WAN) es distinto de usar Linux como equipo de escritorio, como servidor de archivos y de impresión o como cortafuegos.

Las primeras veces, los administradores se confunden al configurar el servidor, debido a que no están acostumbrados con los términos y los conceptos. No tendrá la interfaz gráfica del sistema X Window y tendrá que usar comandos en lugar de hacer clic sobre iconos. Su trabajo se hará en modo consola, desde la interfaz de línea de comandos.

Nota: Como parte de nuestra estrategia para enseñar administración, le enseñaremos cómo disponer de una herramienta basada en Web para el sistema en el próximo capítulo (los proveedores de servicios usan esta herramienta basada en Web para gestionar los servidores Linux que alquilan a los clientes). Por lo que no estará limitado a una pantalla en blanco y negro.

Si sigue las instrucciones de este capítulo, obtendrá un marco para hospedar sitios Web que podrá adaptar a otros propósitos. Su sistema dispondrá de:

- Un servidor Web (Apache 2.0.x).
- Un servidor de correo (Postfix).
- Un servidor DNS (BIND 9).
- Un servidor FTP (ProFTPD).
- Agentes de transferencia de correo (POP3/POP3/IMAP/IMAP).
- Analizador para estadísticas Web.

Aunque hay muchas formas de configurar un servidor Web remoto, seguir las instrucciones que proporcionamos es una buena base para iniciarse en Linux. Una vez que haya aprendido a instalar, tendrá la posibilidad de configurar los servidores a su medida.

Nota: Durante el proceso de instalación, probablemente verá comandos y conceptos con los que no está familiarizado. Le guiaremos para que no introduzca datos sin pensarlo antes. Aunque intentaremos explicar todo lo posible el proceso de instalación, probablemente no quedará satisfecho con la información de este capítulo.

Es difícil para alguien retener información compleja en una primera lectura. Así es que, aunque aconsejarle los comandos a introducir puede parecer ineficiente, le permitirá retener suficiente información sobre la materia que más tarde podrá asimilar. Cubriremos cada tema con gran detalle en los siguientes capítulos, y esta experiencia le ayudará a seguir el curso que propone el libro.

Usted y su servidor se encuentran en la puerta de un nuevo mundo, el de Linux, así es que comencemos.

Requisitos del servidor

Puede usar casi cualquier distribución de Linux para configurar un servidor Web. En este ejercicio, usaremos Debian. Hemos elegido Debian porque queremos usar una distribución estable de Linux. Las principales distribuciones comerciales (Red Hat Linux y Novell Suse Linux) tienen precios que los alejan de la mayoría de los usuarios, pero puede obtener Debian gratis. También Red Hat y SUSE usan herramientas de gestión propietarias que crean dificultades para transferir conocimiento Linux.

Puede aprender más sobre el comportamiento estándar de Linux usando Debian que usando SUSE o Red Hat.

Para configurar un Servidor de Internet sobre Linux, necesitará una conexión a Internet y una dirección IP estática. Si no puede conseguir una dirección IP estática, puede configurar el sistema con la dirección cedida por su ISP y configurarla estáticamente. Asegúrese de que sabe el tiempo que dura dicha dirección, por si necesita cambiarla mientras el sistema está ejecutándose.

También necesitará un ordenador, con al menos una CPU Pentium II con un mínimo de 256 MB de RAM y 10 GB de disco duro. Obviamente, una CPU más moderna y memoria adicional mejorarán el rendimiento.

Este capítulo se basa en la versión estable de Debian. Recomendamos usar un CD con el núcleo NetInstall. El sitio Web de Debian (`http://www.debian.org`) ofrece imágenes de CD descargables.

Instalando Debian

Asumimos que sabe cómo hacer una instalación de red de Linux. Sólo necesitará unos cuantos pasos para configurar su instalación base.

Después de que inicie el equipo con el CD-ROM de arranque de Debian, verá una pantalla de autentificación. Asegúrese de introducir linux26 para así trabajar sobre la versión 2.6 del kernel que es más reciente que la antigua versión 2.4.

El programa de instalación le guiará a través de una serie de pantalla de instalación. Cuando llegue a la pantalla Configure su red, Debian primero sugiere configurar su red con DHCP. Puede hacerlo si dispone de DHCP. Si no es así, Debian por defecto le mostrará una pantalla que le permite configurar la red de manera manual. Se le preguntará el nombre de host que quiere dar al servidor, un nombre de dominio, una pasarela, una dirección IP, una máscara de red y un nombre de servidor. Si tiene registrado un dominio y una dirección IP estática, está listo para continuar. Si no tiene registrado un nombre de dominio, necesitará uno.

Ahora que ya ha configurado su red, puede continuar con las tareas de instalación hasta que complete la instalación básica. El script de instalación de Debian le guiará a través de las siguientes secciones.

Seguidamente, llegará a la pantalla de particionamiento de disco. Para los propósitos de este libro, cree sólo una partición con el punto de montaje / (barra

invertida) y una partición swap. Elija la opción que pone todos los archivos en una partición. Finalmente, elija la opción de finalizar el particionamiento y escriba los resultados en el disco.

Nota: Puede obtener un nombre de dominio desde multitud de fuentes desde 3 USD. Busque en Internet las palabras "registro de dominios". Verá una lista de lugares. Muchos vendedores ofrecen sus servicios a precios bajos y otros los ofrecen gratis. Necesitará dos servidores DNS registrado para obtener un nombre de dominio inicialmente. También puede usar otro nombre de dominio si no tiene un servidor físico que le ofrezca un servidor de nombres secundario. Cada dominio que registre necesita un servidor DNS primario y una copia o un servidor DNS secundario.

Nota: La instalación básica de Debian que estamos usando tiene dos secciones distintas. La primera instala lo que algunos llaman el motor GNU/Linux, que le permite arrancar el disco duro y obtener una pantalla de root. Además transfiere archivos desde el CD-ROM hasta el disco duro.

Una vez que la primera sección termina, le pide que extraiga el CD-ROM que usó en la instalación. Desde este punto en adelante, la instalación continúa usando los archivos almacenados en el disco duro.

Ahora continúan unas cuantas pantallas de instalación, que le solicitarán que reinicie el núcleo para finalizar la instalación.

Después de reiniciar, Debian le pedirá que añada un usuario sin privilegios durante la instalación, que le permitirá autentificarse y usar el comando su para convertirse en root. Por razones de seguridad, los administradores de sistemas tienen la práctica común de no entrar en el sistema como root, a menos que necesiten recuperar el sistema tras un fallo.

Ponga el nombre de Administrador y el ID admin. No use la misma contraseña para admin y para root. Usaremos el usuario de ID admin en otros capítulos.

Cuando llegue a la pantalla de selección de software Debian, ponga el cursor en el recuadro Servidor de correo, pulse la barra de espacio y permita que el sistema instale los paquetes por defecto hasta que llegue a la opción donde se vea cliente libc.

Debería instalar el cliente libc con el buzón de correo de Unix en lugar del soporte maildir. Los buzones de correo Unix mantienen todo el correo en un archivo simple, mientras que maildir mantiene cada mensaje en un archivo distinto. Los buzones de correo de Unix son más fáciles de configurar, por lo que de momento será con los que trabajemos.

Debian también querrá que configure Exim como agente de transferencia de correo (MTA), pero no lo haga. Sustituiremos Exim con Postfix un poco más

tarde en este capítulo. Mientras tanto, cuando llegue a la pantalla que dice Configurando Exim v4, elija la opción Sin configuración. Luego elija Sí cuando el script le pregunte "¿Realmente quiere dejar el sistema de correo sin configurar?"

Finalmente, en la última pantalla que tiene que ver con Exim, introduzca el nombre **admin** y el correo electrónico del root y del administrador de correo.

MTA: Sendmail y Alternativas

El proceso de instalación por defecto de Debian gira en torno a Exim, aunque otras distribuciones Linux usan Sendmail por defecto. Sendmail es de hecho el estándar MTA, como lo prueban las primeras distribuciones de Linux. Casi todos los procesos en Linux que están relacionados con el correo involucran archivos de configuración Sendmail, y la mayoría de las aplicaciones de software libre esperan que Sendmail esté instalando en el sistema operativo.

Probablemente sea una locura reemplazar Sendmail por otro MTA. Si instala Red Hat, Sendmail se instala por defecto. Sin embargo, Red Hat y Fedora vienen con un programa que permite al usuario cambiar a Postfix, aunque debe hacerse de manera manual.

Los gestores del proyecto Debian eligieron Exim como el MTA por defecto porque su creador le otorgó la licencia pública GPL. Al igual que Postfix, Exim es un sustituto para Sendmail.

Hoy en día, la práctica común es usar Postfix, por muchas razones que más tarde explicaremos. De todas formas usted no estropeará su sistema al reemplazar Exim por Postfix. De hecho, podrá descargar Postfix desde los repositorios de Debian.

Autentificándose remotamente

Cuando finalice la instalación, debería acceder al sistema desde la consola remota de su escritorio. Recomendamos que realice las labores de administración desde otro sistema (incluso desde un portátil), porque un servidor seguro normalmente se ejecuta en lo que se llama modo *headless*, es decir, que puede ejecutar aplicaciones sin monitor y sin teclado. Acostúmbrese a administrar su servidor de esta forma, como si estuviera en un sitio en producción. En la máquina remota necesitará solamente un cliente SSH, incluido en todas las distribuciones Linux y que puede descargarse para otros sistemas operativos.

La siguiente salida se produce cuando usa por primera vez SSH en su nuevo servidor Linux:

```
$ ssh admin@server1.centralsoft.org
The authenticity of host 'server1.centralsoft.org (70.253.158.42)' can't
be established.
```

```
RSA key fingerprint is 9f:26:c7:cc:f2:f6:da:74:af:fe:15:16:97:4d:b3:e6.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'server1.centralsoft.org,70.253.158.42' (RSA)
to the list of known hosts.
Password: enter password for admin user here
Linux server1 2.6.8-2-386 #1 Thu May 19 17:40:50 JST 2005 i686 GNU/Linux

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Dec 25 19:07:38 2005 from 70.255.197.162
admin@server1:~$
```

Llegados a este punto, ha establecido una conexión remota y puede realizar tareas como si estuviera delante del monitor de su servidor. Si lo desea, puede quitar cualquier monitor, teclado y ratón que haya conectado al servidor.

Configurando la red

Si usó DHCP durante la instalación de Debian, ahora debería configurar su servidor con una dirección IP estática, por lo que podrá realizar el test necesario como se explica más tarde en este mismo capítulo. Si usted tuviera una dirección IP pública configurada como estática, puede saltar a la siguiente sección.

Si instaló Debian con un cliente DHCP para su router o para su proveedor de acceso a Internet, necesitará reconfigurar la red. Esto supone una lección valiosa si quiere explorar la configuración de red de Linux.

Para cambiar la configuración y usar una dirección IP estática, necesitará convertirse en root y editar el archivo `/etc/network/interfaces` de acuerdo a sus necesidades. Como ejemplo, vamos a usar la dirección IP 70.153.258.42.

Nuestro archivo de configuración comienza así:

```
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
# The loopback interface
auto lo
iface lo inet loopback
# The first network card - this entry was created during the Debian
# installation
# (network, broadcast, and gateway are optional)
# The primary network interface
iface eth0 inet dhcp
```

Para añadir la dirección IP 70.153.258.42 a la interfaz eth0, debe cambiar el archivo para que quede así (deberá preguntar por esta información a su ISP):


```
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)
# The loopback interface
auto lo
iface lo inet loopback
# The first network card - this entry was created during the Debian
# installation
# (network, broadcast, and gateway are optional)
auto eth0
iface eth0 inet static
   address 70.153.258.42
   netmask 255.255.255.248
   network 70.153.258.0
   broadcast 70.153.258.47
   gateway 70.153.258.46
```

Después de editar `/etc/network/interfaces` file, reinicie la red introduciendo esto:

```
# /etc/init.d/networking restart
```

Necesitará editar `/etc/resolv.conf` y añadir nameservers para resolver los nombres de host relacionados con la correspondiente dirección IP. Aunque no hemos configurado nuestro propio servidor de nombres, lo haremos más tarde en este capítulo. En este punto, simplemente configuraremos un servidor DNS mínimo. Los otros servidores de nombre deberían especificar la dirección IP de los servidores DNS ofrecidos por tu ISP. Nuestro `resolv.conf` quedaría así:

```
search server
nameserver 70.153.258.42
nameserver 70.253.158.45
nameserver 151.164.1.8
```

Nota: Asegúrese de usar servidores DNS que funcionen con su dominio; en caso contrario, su servidor DNS no indicará quién es la autoridad para su dominio.

Ahora edite `/etc/hosts` y añada su dirección IP:

```
127.0.0.1       localhost.localdomain     localhost   server1
70.153.258.42   server1.centralsoft.org   server1
```

Nota: Ignore la información IPv6 en el archivo `/etc/hosts`. Le mostraremos cómo configurar el servidor IPv6 más adelante.

Ahora configure el nombre de host, introduzca estos comandos:

```
# echo server1.centralsoft.org > /etc/hostname
# /bin/hostname -F /etc/hostname
```

Necesitará usar los mismos comandos independientemente de cómo configurara la red durante la instalación, sustituyendo su nombre de dominio por `server1.centralsoft.org`. Luego, verifique que ha configurado su nombre de host correctamente ejecutando el comando hostname:

```
~$ hostname
server1
~$ hostname -f
server1.centralsoft.org
```

Si obtiene este resultado, está listo para ir a la siguiente sección. De no ser así, revise el archivo `/etc/hostname`. Debería ser parecido a esto:

```
#less /etc/hostname
server1
```

Oops. Debería leer atentamente `server1.centralsoft.org`. Puede modificarlo ahora.

Cambiando los paquetes por defecto de Debian

Comenzamos con los paquetes Debian que los desarrolladores colocan en su distribución por defecto. Como indicamos anteriormente, necesitamos hacer algunos cambios, sobre todo para hacer uso de Posftix. Aunque podría pensar que estamos relegando a un segundo plano el trabajo hecho por el equipo de Debian, este no es el caso.

El equipo Debian ha elegido instalar, por defecto, servicios apropiados para una LAN como el NetworkFile System (NFS). Pero nosotros estamos poniendo el servidor en Internet, por lo que eliminaremos NFS y otros servicios, mientras que añadiremos otros como OpenSSL.

Para obtener todos los archivos necesarios para este capítulo, ejecute el siguiente comando:

```
# apt-get install wget bzip2 rdate fetchmail libdb3++-dev \
unzip zip ncftp xlispstat libarchive-zip-perl \
zlib1g-dev libpopt-dev nmap openssl lynx fileutils
```

Verá a Debian descargar archivos en su consola. Luego, la descarga finalizará y verá un mensaje que le preguntará si desea continuar:

```
0 upgraded, 42 newly installed, 0 to remove and 0 not upgraded.
Need to get 12.2MB of archives.
After unpacking 35.8MB of additional disk space will be used.
Do you want to continue? [Y/n]
```


Pulsando Y completará la instalación de los archivos adicionales. Luego, habrá que eliminar los servicios que no vaya a usar. Ejecute el siguiente comando y verá una salida como la siguiente:

```
# apt-get remove lpr nfs-common portmap pidentd pcmcia-cs \
pppoe pppoeconf ppp pppconfig
Reading Package Lists... Done
Building Dependency Tree... Done
Package pcmcia-cs is not installed, so not removed
The following packages will be REMOVED:
   lpr nfs-common pidentd portmap ppp pppconfig pppoe pppoeconf
0 upgraded, 0 newly installed, 8 to remove and 0 not upgraded.
Need to get 0B of archives.
After unpacking 3598kB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 22425 files and directories currently installed.)
Removing lpr ...
Stopping printer spooler: lpd .
Removing nfs-common ...
Stopping NFS common utilities: statd.
Removing pidentd ...
Removing portmap ...
Stopping portmap daemon: portmap.
Removing pppoeconf ...
Removing pppoe ...
Removing pppconfig ...
Removing ppp ...
Stopping all PPP connections...done.
```

Nota: Asegúrese de revisar los comandos que introduzca. Si comete un error, Debian le indicará que no puede encontrar el archivo en cuestión. En este caso, simplemente reintroduzca `apt-get`, especificando sólo el nombre del paquete. Desde que haga cambios en el paquete de la base de datos, necesita cambiar los scripts que se ejecutan cuando se arranca. Use los siguientes comandos para modificar los scripts de inicio:

```
# update-rc.d -f exim remove
Removing any system startup links for /etc/init.d/exim ...
# update-inetd --remove daytime
# update-inetd --remove telnet
# update-inetd --remove time
# update-inetd --remove finger
# update-inetd --remove talk
# update-inetd --remove ntalk
# update-inetd --remove ftp
# update-inetd --remove discard
```

Ahora necesita reiniciar `inetd`, que es el proceso servidor para los servicios estándares de Internet. `inetd` normalmente se lanza al arrancar, pero como ha

cambiado los servicios del sistema, necesita reiniciarlo para que pueda descubrir los nuevos servicios del archivo de configuración. El comando `inetd` acepta un parámetro que apunta al archivo de configuración listando los servicios que ofrece. Pero si no se proporciona un parámetro por línea de comandos, `inetd` lee la información de configuración del archivo `/etc/inetd.conf`, que sirve para nuestros propósitos. El comando de actualización de `inetd` almacenó nuestros cambios en este archivo. Para reiniciar `inetd` usando el archivo de configuración por defecto, introduzca:

```
# /etc/init.d/inetd reload
```

Verá el siguiente mensaje en su consola:

```
Reloading internet superserver: inetd
```

Configurando cuotas

El servidor Web Apache ofrece a Linux la posibilidad de ofrecer alojamiento virtual, es decir, su servidor puede alojar varios sitios Web con nombres de dominio que difieren del nombre del servidor físico. En el archivo de configuración del servidor Web, puede definir diferentes dominios usando cláusulas de alojamiento virtual. Por ejemplo, incluso aunque el dominio usado en este libro es `centralsoft.org`, podríamos tener `mothersmagic.com`, `wildbills.info` u otro dominio que registremos y usemos con la misma dirección IP. Cubriremos este concepto con detalle más adelante. Por ahora, sólo piense en la dirección IP como el número de teléfono de una casa donde viven diferentes personas. Cuando el navegador accede al puerto 80, llega hasta el dominio que usted ha configurado. Linux ofrece medios para gestionar el uso del disco para múltiples dominios gracias a una facilidad llamada cuotas. Originalmente, Unix ofrecía cuotas para las cuentas de usuario para que no ocuparan demasiado espacio en el servidor. Por ejemplo, si tuviera 50 usuarios compartiendo espacio en disco en un servidor de archivos, sin un sistema de cuotas, un usuario podría llenar el disco, provocando que las aplicaciones de todos los usuarios no pudieran guardar más datos.

La facilidad para las cuotas obliga a los usuarios a mantenerse dentro de los límites de consumo marcados, evitando la posibilidad de que consuman espacio ilimitado en disco. El sistema lleva en cuenta la cuota por usuario y por sistema de archivos. Si tiene más de un sistema de archivos donde los usuarios pueden crear archivos, configure la facilidad para cada sistema de archivos de manera separada.

Puede usar el mismo sistema de cuota para limitar el espacio reservado para un dominio de su host. Varias herramientas le permiten administrar y automatizar la política de cuotas de su sistema. En esta parte de la configuración del servidor, añadirá una facilidad para cuotas que podrá usarse después. Primero, instala los paquetes de cuotas usando apt-get:


```
# apt-get install quota quotatool
```

Le aparecerá una pregunta como esta:

```
Enable this option if you want the warnquota utility to be run daily
to alert users
when they are over quota.
Send daily reminders to users over quota?
<Yes> <No>
```

Aquí debe escoger <No>.

Debian instalará y configurará los dos paquetes, pero usted tendrá que editar `/etc/fstab` para activar las cuotas en los sistemas de archivos que desee. Debido a que nuestro sistema tiene una única partición para todos los archivos de usuario, puede añadir las opciones usrquota y grpquota a la partición con el punto de montaje /:

```
# /etc/fstab: static filesystem information.
#
# <filesystem> <mount point> <type>  <options>         <dump>  <pass>
proc           /proc         proc    defaults          0       0
/dev/sda1      /             ext3    defaults,errors=remountro,
usrquota,grpquota 0          1
/dev/sda5      none          swap    sw                0       0
/dev/hdc       /media/cdrom0 iso9660 ro,user,noauto    0       0
/dev/fd0       /media/floppy0 auto   rw,user,noauto    0       0
```

Ahora ejecute los siguientes comandos para añadirlos al directorio raíz:

```
# touch /quota.user /quota.group
# chmod 600 /quota.*
# mount -o remount /
# quotacheck -avugm
```

El kernel de Linux normalmente tiene soporte por defecto para las cuotas. El kernel ve las opciones de cuota en `/etc/fstab` y comprueba `quota.user` y `quota.group` para determinar si los usuarios o los grupos tienen límites en su espacio en disco.

Ahora verá lo siguiente en su consola:

```
quotacheck: Scanning /dev/hda1 [/] done
```

Ahora verá un mensaje en su consola parecido a este:

```
quotacheck: Checked 1912 directories and 28410 files
```

Ahora puede ejecutar el próximo comando:

```
# quotaon -avug
```

Verá los siguientes mensajes:

```
/dev/hda1 [/]: group quotas turned on
/dev/hda1 [/]: user quotas turned on
```

¿Ha comprendido lo que ha pasado? Esta secuencia activó las cuotas en el sistema. Puede revisar las páginas de ayuda para cuotas si cree que no lo ha comprendido. El servidor ahora está listo para usar las cuotas.

Ofreciendo servicios de nombre de dominio

En el capítulo siguiente, aprenderemos cómo manejar nombres de dominio para su servidor y para dominios virtuales que residan en su sistema. Por ahora, estableceremos una configuración mínima para BIND, el servidor DNS ubicuo.

Debian ofrece una versión estable de BIND en sus repositorios. Nosotros instalaremos y configuraremos BIND y lo aseguraremos en un entorno chroot, es decir, donde no será posible acceder a los archivos que están fueran de su propio árbol de directorios. Esta es una técnica de seguridad importante. El término chroot se refiere al truco de cambiar el sistema de archivos raíz (el directorio /) que ve un proceso, por la que la mayor parte del sistema no puede acceder a él. También configuraremos BIND para ejecutarlo como un usuario que no sea root. De esta forma, si alguien consigue el acceso a BIND, no podrá obtener privilegios de root o ser capaz de controlar otros procesos. Para instalar BIND en su servidor Debian, ejecuta este comando:

```
# apt-get install bind9
```

Debian baja y configura el archivo como un servidor de Internet. Podrá ver el siguiente mensaje en su pantalla:

```
Setting up bind9 (9.2.4-1)
Adding group 'bind' (104)
Done.
Adding system user 'bind'
Adding new user 'bind' (104) with group 'bind'.
Not creating home directory.
Starting domain name service: named.
```

Nota: Verá salidas similares cuando instale o elimine otros servicios con la utilidad apt-get.

Para poner BIND en un entorno seguro, cree un directorio donde el servicio pueda ejecutarse sin tener que estar expuesto a los otros procesos. También podrá ejecutarlo como usuario sin privilegios, pero sólo el root podrá acceder al directorio.

Primero pare el servicio ejecutando el siguiente comando:

```
# /etc/init.d/bind9 stop
```

Edite el archivo `/etc/default/bind9` por lo que el demonio se ejecutará como usuario sin privilegios, el directorio será `/var/lib/named`. Cambie la línea:

```
OPTS="-u bind"
```

para que ponga:

```
OPTIONS="-u bind -t /var/lib/named"
```

Para ofrecer un entorno completo para ejecutar BIND, cree los directorios necesarios en `/var/lib`:

```
# mkdir -p /var/lib/named/etc
# mkdir /var/lib/named/dev
# mkdir -p /var/lib/named/var/cache/bind
# mkdir -p /var/lib/named/var/run/bind/run
```

Luego, mueva el directorio config desde `/etc` a `/var/lib/named/etc`:

```
# mv /etc/bind /var/lib/named/etc
```

Ahora cree un enlace simbólico al nuevo directorio config desde la localización antigua, para evitar problemas cuando BIND se actualice en el futuro:

```
# ln -s /var/lib/named/etc/bind /etc/bind
```

Cree dispositivos null y random para que los use BIND y establezca los permisos de los directorios:

```
# mknod /var/lib/named/dev/null c 1 3
# mknod /var/lib/named/dev/random c 1 8
```

Luego, cambie los permisos y la autoría de los archivos:

```
# chmod 666 /var/lib/named/dev/null /var/lib/named/dev/random
# chown -R bind:bind /var/lib/named/var/*
# chown -R bind:bind /var/lib/named/etc/bind
```

También necesitará cambiar el script de inicio `/etc/init.d/sysklogd` para que pueda ver mensajes en los logs del sistema. Cambie la línea:

```
SYSLOGD=""
```

para que ponga:

```
SYSLOGD="-a /var/lib/named/dev/log"
```

Ahora reinicie el proceso de logging con este comando:

```
# /etc/init.d/sysklogd restart
```

Verá el siguiente mensaje:

```
Restarting system log daemon: syslogd.
```

Finalmente, ejecute BIND:

```
# /etc/init.d/bind9 start
```

Compruebe `/var/log/syslog` por si hay errores. Puede moverse por el archivo usando:

```
# less /var/log/syslog
```

Puede asegurarse que BIND se arrancó con éxito si puede ver:

```
Starting domain name service: named.
```

Ahora, comprobemos si named está funcionando sin problemas. Ejecute este comando, debería ver un resultado como el siguiente:

```
server1:/home/admin# rndc status

number of zones: 6
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
server is up and running
server1:/home/admin#
```

Si DNS no está funcionando correctamente, en su lugar podrá ver algo como lo siguiente:

```
server1:~# rndc status
rndc: neither /etc/bind/rndc.conf nor /etc/bind/rndc.key was found
server1:~#
```

Afortunadamente, nuestro sistema DNS está funcionando correctamente.

Por el momento, no hemos configurado nuestros archivos de zona primaria ni hemos configurado DNS para que actúe como servidor caché, el cual almacena en caché cada visita a una página Web. Le mostraremos cómo configurar el servidor DNS primario y secundario en el capítulo siguiente.

Aunque muchas personas creen que carece de importancia, administrar DNS es crucial porque muchos otros servicios dependen de él. Ya verá como DNS es un componente crítico de casi todos los servicios de Internet que su sistema ofrece.

Añadiendo una base de datos relacional: MySQL

Los sitios y los servicios Web usan bases de datos relaciones para colocar objetos en las páginas Web. Esto permite un acceso rápido a la hora de atender las peticiones. Los navegadores Web pueden realizar 30 peticiones de una vez, incrementando las cargas en la CPU, en la memoria y en el acceso a disco.

Las bases de datos relaciones, en combinación con un servidor Web, puede construir de manera eficiente páginas Web al vuelo.

No vamos a cubrir el complejo asunto de la administración de bases de datos en este libro. Sin embargo, los administradores de sistemas Linux a menudo se encuentran con que los desarrolladores esperan contar con bases de datos para su desarrollo, por lo que le mostraremos la manera de configurar su servidor Linux con una de las bases de datos de código abierto más populares: MySQL. Para hacer efectivo el uso de la base de datos, necesitará aprender a:

1. Instalar e iniciar MySQL.
2. Crear un usuario root para MySQL.
3. Crear un usuario MySQL normal, que será usado por las aplicaciones para acceder a la base de datos.
4. Realizar copias de seguridad y restauraciones de la base de datos.

Para instalar correctamente el servidor de bases de datos, un programa cliente adecúado para las tareas de administración y la biblioteca necesaria por ambos, use este comando:

```
# apt-get install mysql-server mysql-client libmysqlclient12-dev
```

Debian descargará MySQL desde sus repositorios y comenzará el proceso de instalación. Verá los siguientes mensajes:

```
Install Hints
MySQL will only install if you have a NON-NUMERIC hostname that is
resolvable via the /etc/hosts file. E.g. if the "hostname" command
returns "myhostname" then there must be a line like "10.0.0.1
myhostname".
A new mysql user "debian-sys-maint" will be created. This mysql account
is used in the start/stop and cron scripts. Don't delete.
```

```
Please remember to set a PASSWORD for the MySQL root user! If you use a
/root/.my.cnf, always write the "user" and the "password" lines in
there, never only the password!
See /usr/share/doc/mysql-server/README.Debian for more information.
<Ok>
```

Desde el punto de vista de la administración, MySQL es comparable a Linux: cada usuario root tiene el control sobre todo y puede conceder o denegar privilegios a los otros usuarios. El root MySQL no tiene nada que ver con el root de Linux, lo único que tienen en común es el nombre. Cree el usuario root de MySQL introduciendo:

```
# mysqladmin -u root password 'pword'
```

Elija una cadena que sea difícil de adivinar como contraseña (pwrod). Cuando quiera administrar SQL en el futuro, deberá introducir el siguiente comando y teclear su contraseña:

```
# mysql -u root -p
Enter password:
```

Intente asegurarse de que el cliente y el servidor están trabajando y que pueden acceder al servidor. Debería ver en la salida de la consola algo similar a lo siguiente:

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 14 to server version: 4.0.24_Debian-10-log
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

Teclee `/q` o `quit` para salir.

Ya que el servidor MySQL se está ejecutando, puede ejecutar netstat –tap y ver algo como esto:

```
tcp   0  0 localhost.localdo:mysql *:*   LISTEN   2449/mysqld
```

MySQL es accesible desde localhost (127.0.0.1) en el puerto 3306. Si no ve esta línea, edite `/etc/mysql/my.cnf` (el archivo de configuración que el cliente y el servidor comprueban para obtener los parámetros) y añada un símbolo # para comentar la línea skip-networking:

```
#skip-networking
```

Si quiere que MySQL escuche en todas las direcciones IP disponibles, edite `/etc/mysql/my.cnf` y comente la línea bind-address = 127.0.0.1:

```
#bind-address     = 127.0.0.1
```


Si ha editado `/etc/mysql/my.cnf`, reinicie MySQL usando este comando:

```
# /etc/init.d/mysql restart
```

Esta sección no ha cubierto todas las funciones de las bases de datos que los desarrolladores esperan de usted. MySQL está ahora configurado para ejecutarse en su servidor, no obstante, esto es suficiente para que pueda dar los siguientes pasos. Haremos más cosas con MySQL en próximos capítulos.

Configurando el correo de manera segura con Postfix, POP3 e IMAP

En esta sección, añadiremos agentes de transporte y de entrega de correo electrónico e implementaremos un ligero control sobre el entorno de los sistemas. Demostraremos cómo autentificar a los usuarios de un sistema de correo electrónico y prevenir el acceso fraudulento a aspectos del correo. Durante más de 25 años, Sendmail ha servido como el MTA primario de Internet. Muchas aplicaciones escritas para Linux esperan que Sendmail se esté ejecutando en el servidor. A pesar de que fue escrito antes de que Internet se abriera al público, Sendmail tiene muchos problemas de seguridad que se listan en la lista Common Vulnerabilities and Exposures (CVE) alojada en `http://cve.mitre.org`.

Afortunadamente, otros MTA han emergido para ocupar el lugar de Sendmail. El principal problema de estos MTA es que las aplicaciones esperan que sea Sendmail el que está presente en el servidor Linux. Para solucionar esto, algunos MTA como Postfix o Exim deben ser capaces de hacer creer a las aplicaciones que ellos son Sendmail. Son sustituciones de andar por casa y pueden hacer que el sistema se ejecute en modo Sendmail.

Postfix es nuestro sustituto preferido para Sendmail. Postfix es más rápido que Sendmail, tiene una arquitectura más segura y modular y ofrece muchas funciones requeridas por un proveedor de grandes volúmenes de correo. Postfix no soporta protocolos obsoletos, sino que usa el Protocolo Simple de Transporte de Correo (SMTP), y es el que tiene el menor número de incidencia en la lista CVE. Por todas estas razones, usaremos Postfix en lugar de Sendmail como MTA.

El correo electrónico seguro involucra mantener fuera del servidor a todos los usuarios sin autorización (por lo que no pueden usarlo para enviar correo anónimo), asegurándose de que nadie puede suplantar a los usuarios legítimos y protegiendo el contenido de cada correo recibido o despachado.

Una seguridad débil hace más fácil que los impostores suplanten usuarios. Para proteger la autentificación, instalaremos con Postfix la *Transport Layer Security* (TLS), un protocolo mejor que el conocido *Secure Sockets Layer* (SSL). Esto

evita enviar contraseñas en texto claro desde un cliente de correo electrónico a un servidor.

También queremos que los usuarios se autentifiquen y accedan al servidor de correo. Para conseguirlo, emplearemos el *Simple Authentication Security Layer* (SASL). Esto crea una extensión (ESMTP) que permite a un cliente SMTP autentificar el servidor.

Instalar los paquetes es necesario para Postfix y los otros componentes de correo, introduzca:

```
# apt-get install postfix postfix-tls libsasl2 sasl2-bin \
libsasl2-modules ipopd-ssl uw-imapd-ssl
```

A medida que Debian instale los paquetes, presentará algunas pantallas que le preguntarán algunas cuestiones.

Cuando vea la pantalla de configuración de ipopd mostrada en la figura 2.1, seleccione pop3 y pop3s.

```
Configuring ipopd
pop2 = POP 2 on TCP port 109 with TLS support.  This is obsolete.

pop3  = POP 3 on TCP port 110 with TLS support.

pop3s = POP 3 on TCP port 995 with SSL support.

If in doubt, choose pop3 and pop3s.

Which ports should the server listen on?

    [ ] pop2
    [*] pop3
    [*] pop3s

                         <Ok>
```

Figura 2.1. Pantalla de configuración de correo de Debian.

Luego, verá una pantalla parecida a la de figura 2.2, donde debería seleccionar **<No>** para conseguir flexibilidad a la hora de enrutar los puertos en caso de lo que necesite después. Los puertos por defecto trabajan aquí porque usan TLS y el demonio SASL.

La figura 2.3 es informativa; el instalador Debian está indicando qué opciones tiene para la configuración de correo. Haga clic en **OK** para pasar a la pantalla de la figura 2.4, que le permite escoger una opción. Para nuestro propósito, escogeremos Internet Site, porque usaremos SMTP para todo el tráfico, tanto para la LAN como para Internet. Debian ofrecerá el tipo de archivo de configuración

que mejor se ajuste a nuestras necesidades. Más tarde podremos añadir esto a la configuración por defecto.

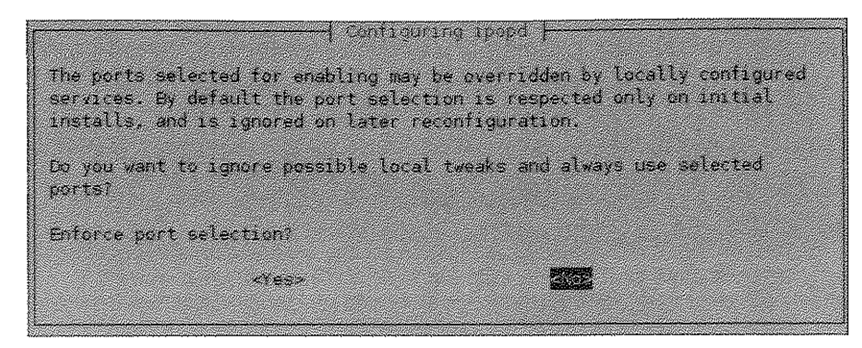


Figura 2.2. Dejando los puertos por defecto para el correo.

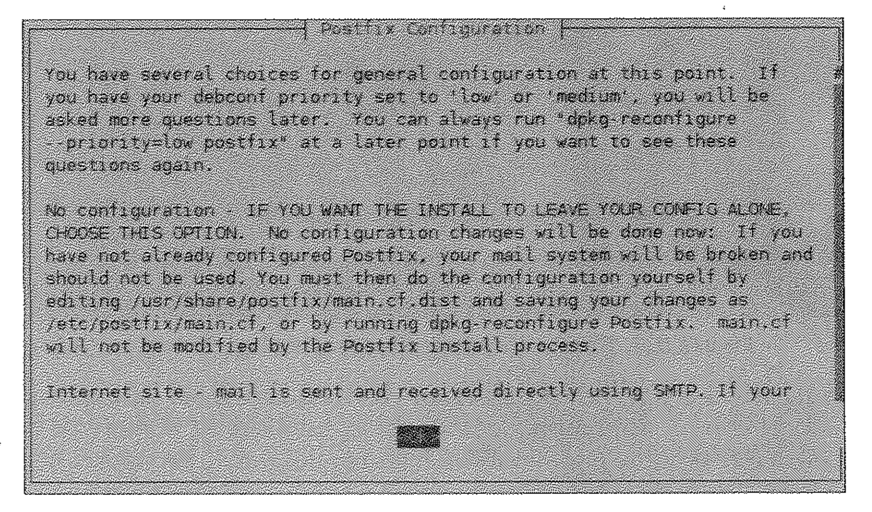


Figura 2.3. Opciones de configuración Postfix.

Cuando configure Postfix para ejecutar el correo, funcionará como un agente de transferencia de correo estándar. No escoja la opción de la figura 2.4 para usar otro servidor de correo como smarthost. En otras palabras, su sistema será la autoridad de correo de su dominio. Si ha usado otro servidor (un portal popular o un ISP) para enviar y recibir correo en el pasado, su servidor realizará estas tareas ahora.

Luego, en la pantalla mostrada en la figura 2.5, responda NONO. Postfix creará su propio archivo de alias.

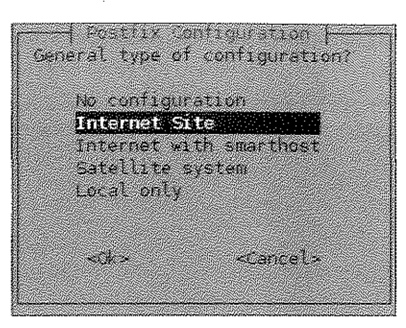


Figura 2.4. Seleccionando el sitio de Internet desde el menú de configuración.

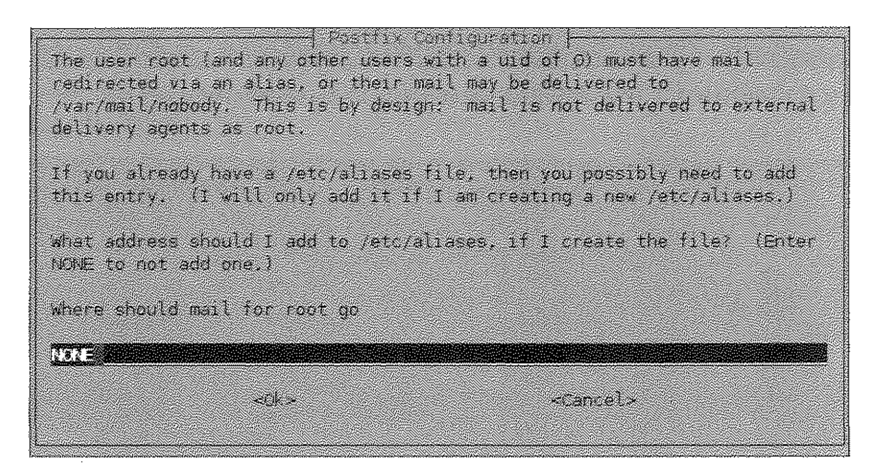


Figura 2.5. Opción para usar una cuenta alias existente.

En las figuras 2.6 y 2.7, el configurador Postfix quiere saber para quién acepta y despacha el correo. El nombre de dominio es también el "nombre de correo". Postfix usará este nombre para verificar el correo dirigido al servidor. Cuando alcance las pantallas mostradas en las figuras 2.6 y 2.7, tendrán valores por defecto en las cajas azules de texto. Puede aceptar la figura 2.6 tal y como le mostramos.

Nota: `centralsoft.org` es el nombre de dominio que usamos en este libro, asegúrese de sustituirlo por su nombre de dominio.

En la figura 2.7, notará que hay dos comas después del nombre `localhost.centralsoft.org`. Elimine la segunda coma.

En la figura 2.8, el configurador de Postfix necesita información sobre la actualización síncrona. Cubriremos la administración de servidores de correo con gran detalle más adelante. Por ahora, responda **<No>** a la pregunta y siga avanzando.

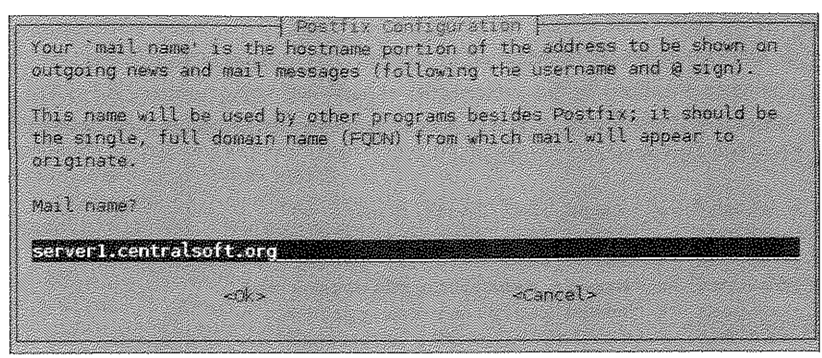


Figura 2.6. Comprobando el nombre de dominio para Postifx.

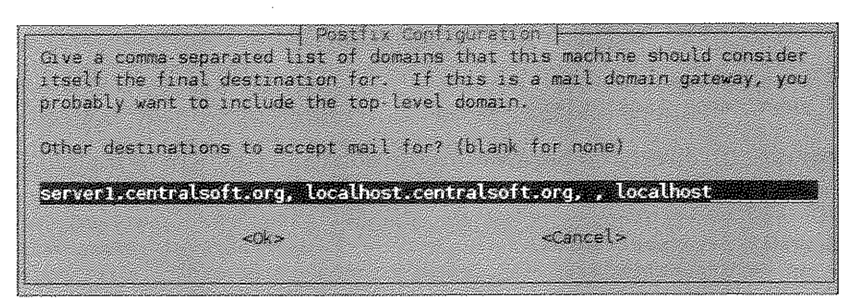


Figura 2.7. Lista interna de dominios usada en Postfix.

Después de que Debian acabe la instalación y vea que la consola vuelve a estar disponible, necesitará poner a trabajar juntos varios componentes de correo. Esto significa que deberá escribir entradas en el archivo de configuración Postfix y generar certificados y claves de encriptación.

Le advertimos sobre esta parte al principio del capítulo. Algunos de estos comandos no tendrán sentido para usted. Pero no se preocupe, le verá el sentido cuando vuelva la vista atrás después de haber completado las tareas de esta sección.

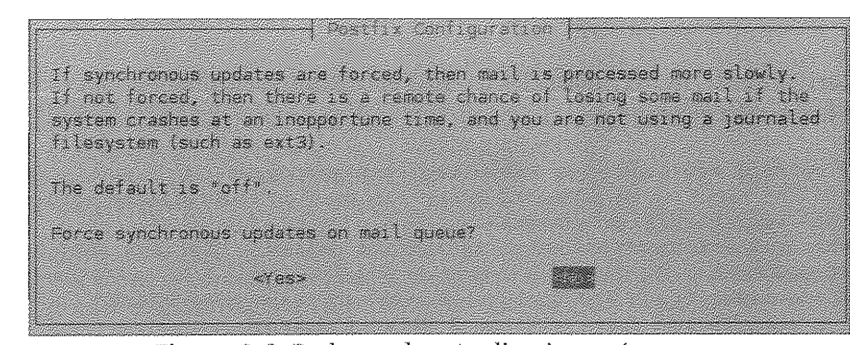


Figura 2.8. Rechazando actualizaciones síncronas.

El comando postconf reside en el directorio `/usr/sbin`. Se usará para escribir el valor de un parámetro Postfix en el archivo de configuración `main.cf`.

Una vez que Posftix se ha instalado y se ha configurado como un servicio Debian, necesitará indicarle a Postfix qué hacer con respeto a la autentificación segura. Usa los siguientes comandos:

```
# postconf -e 'smtpd_sasl_local_domain ='
# postconf -e 'smtpd_sasl_auth_enable = yes'
# postconf -e 'smtpd_sasl_security_options = noanonymous'
# postconf -e 'broken_sasl_auth_clients = yes'
# postconf -e 'smtpd_recipient_restrictions = \
permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination'
# postconf -e 'inet_interfaces = all'
```

Estos comandos escriben texto en el archivo `smtpd.conf`:

```
# echo 'pwcheck_method: saslauthd' >> /etc/postfix/sasl/smtpd.conf
# echo 'mech_list: plain login' >> /etc/postfix/sasl/smtpd.conf
```

Ahora cree un directorio para sus certificados SSL y genere tanto los certificados como las claves de encriptación:

```
# mkdir /etc/postfix/ssl
# cd /etc/postfix/ssl/
# openssl genrsa -des3 -rand /etc/hosts -out smtpd.key 1024
293 semi-random bytes loaded
Generating RSA private key, 1024 bit long modulus
..........................................++++++
.......................................++++++
e is 65537 (0x10001)
Enter pass phrase for smtpd.key:
Verifying - Enter pass phrase for smtpd.key:
```

Luego, ejecute este comando para cambiar los permisos del archivo que contiene la clave OpenSSL RSA:

```
# chmod 600 smtpd.key
```

Posteriormente, genere otra clave y un certificado y cambie las claves existentes por las recién generadas:

```
# openssl req -new -key smtpd.key -out smtpd.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name
or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```



```
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
centralsoft.org
Organizational Unit Name (eg, section) []: web
Common Name (eg, YOUR name) []:
Email Address []:
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []: cso
# openssl x509 -req -days 3650 -in smtpd.csr -signkey smtpd.key -out \
smtpd.crt
Signature ok
subject=/C=US/ST=Texas/L=Dallas/O=centralsoft.org/OU=web/CN=Tom_Adelstein/
emailAddress=tom.adelstein@centralsoft.org
Getting Private key
Enter pass phrase for smtpd.key:
# openssl rsa -in smtpd.key -out smtpd.key.unencrypted
Enter pass phrase for smtpd.key:
writing RSA key
# mv -f smtpd.key.unencrypted smtpd.key
# openssl req -new -x509 -extensions v3_ca -keyout cakey.pem -out \
cacert.pem -days 3650
Generating a 1024 bit RSA private key
.....................++++++
..........................++++++
writing new private key to 'cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name
or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:
Email Address []:
```

Nota: Existe un debate acerca de si al generar un certificado se debería pedir o no información. Nosotros le recomendamos que introduzca la información apropiada a sus circunstancias.

Ahora necesita indicarle a Postfix sus claves y certificados, para ello use los siguientes comandos postconf:

```
# postconf -e 'smtpd_tls_auth_only = no'
# postconf -e 'smtp_use_tls = yes'
# postconf -e 'smtpd_use_tls = yes'
# postconf -e 'smtp_tls_note_starttls_offer = yes'
# postconf -e 'smtpd_tls_key_file = /etc/postfix/ssl/smtpd.key'
# postconf -e 'smtpd_tls_cert_file = /etc/postfix/ssl/smtpd.crt'
# postconf -e 'smtpd_tls_CAfile = /etc/postfix/ssl/cacert.pem'
# postconf -e 'smtpd_tls_loglevel = 1'
# postconf -e 'smtpd_tls_received_header = yes'
# postconf -e 'smtpd_tls_session_cache_timeout = 3600s'
# postconf -e 'tls_random_source = dev:/dev/urandom'
```

El archivo `/etc/postfix/main.cf` debería quedar así:

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete
# version
smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
# Appending .domain is the MUA's job
append_dot_mydomain = no
# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h
myhostname = server1.example.com
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = server1.example.com, localhost.example.com, localhost
relayhost =
mynetworks = 127.0.0.0/8
mailbox_command = procmail -a "$EXTENSION"
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
smtpd_sasl_local_domain =
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_recipient_restrictions =
permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination
smtpd_tls_auth_only = no
smtp_use_tls = yes
smtpd_use_tls = yes
smtp_tls_note_starttls_offer = yes
smtpd_tls_key_file = /etc/postfix/ssl/smtpd.key
smtpd_tls_cert_file = /etc/postfix/ssl/smtpd.crt
smtpd_tls_CAfile = /etc/postfix/ssl/cacert.pem
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
```



```
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
```

Si su archivo coincide con este, puede usar este comando para que los cambios surtan efecto:

```
# /etc/init.d/postfix restart
Stopping mail transport agent: Postfix.
Starting mail transport agent: Postfix.
```

La autentificación se hará a través de saslauthd, un demonio SASL, pero tendrá que cambiar unas cuantas cosas para que funcione correctamente. Debido a que Postfix se ejecuta en modo chroot en `/var/spool/postfix`, introduzca los siguientes comandos:

```
# mkdir -p /var/spool/postfix/var/run/saslauthd
# rm -fr /var/run/saslauthd
```

Ahora, tiene que editar `/etc/default/saslauthd` para activar saslauthd. Elimine el símbolo # en la línea `START=yes` y añada la línea `PARAMS="-m /var/spool/postfix/var/run/saslauthd"` para que el archivo quede así:

```
# This needs to be uncommented before saslauthd will be run automatically
START=yes
PARAMS="-m /var/spool/postfix/var/run/saslauthd"
# You must specify the authentication mechanisms you wish to use.
# This defaults to "pam" for PAM support, but may also include
# "shadow" or "sasldb", like this:
# MECHANISMS="pam shadow"
MECHANISMS="pam"
```

Finalmente, edite `/etc/init.d/saslauthd`. Cambie la línea:

```
dir='dpkg-statoverride --list $PWDIR'
```

por:

```
#dir='dpkg-statoverride --list $PWDIR'
```

Luego, cambie las variables PWDIR y PIDFILE y añada la variable dir cerca del comienzo del archivo:

```
PWDIR="/var/spool/postfix/var/run/${NAME}"
PIDFILE="${PWDIR}/saslauthd.pid"
dir="root sasl 755 ${PWDIR}"
```

El archivo `/etc/init.d/saslauthd` debería quedar así:

```
#!/bin/sh -e
NAME=saslauthd
```

```
DAEMON="/usr/sbin/${NAME}"
DESC="SASL Authentication Daemon"
DEFAULTS=/etc/default/saslauthd
PWDIR="/var/spool/postfix/var/run/${NAME}"
PIDFILE="${PWDIR}/saslauthd.pid"
dir="root sasl 755 ${PWDIR}"
createdir( ) {
# $1 = user
# $2 = group
# $3 = permissions (octal)
# $4 = path to directory
   [ -d "$4" ] || mkdir -p "$4"
   chown -c -h "$1:$2" "$4"
   chmod -c "$3" "$4"
}
test -f "${DAEMON}" || exit 0
# Source defaults file; edit that file to conFigure this script.
if [ -e "${DEFAULTS}" ]; then
   . "${DEFAULTS}"
fi
# If we're not to start the daemon, simply exit
if [ "${START}" != "yes" ]; then
   exit 0
fi
# If we have no mechanisms defined
if [ "x${MECHANISMS}" = "x" ]; then
   echo "You need to conFigure ${DEFAULTS} with mechanisms to be used"
   exit 0
fi
# Add our mechanisms with the necessary flag
PARAMS="${PARAMS} -a ${MECHANISMS}"
START="--start --quiet --pidfile ${PIDFILE} --startas ${DAEMON} --name
   ${NAME} -- ${PARAMS}"
# Consider our options
case "${1}" in
   start)
      echo -n "Starting ${DESC}: "
      #dir='dpkg-statoverride --list $PWDIR'
      test -z "$dir" || createdir $dir
      if start-stop-daemon ${START} >/dev/null 2>&1 ; then
         echo "${NAME}."
      else
         if start-stop-daemon --test ${START} >/dev/null 2>&1; then
            echo "(failed)."
            exit 1
         else
            echo "${DAEMON} already running."
            exit 0
         fi
      fi
      ;;
stop)
```



```
      echo -n "Stopping ${DESC}: "
      if start-stop-daemon --stop --quiet --pidfile "${PIDFILE}" \
         --startas ${DAEMON} --retry 10 --name ${NAME} \
         >/dev/null 2>&1 ; then
            echo "${NAME}."
      else
         if start-stop-daemon --test ${START} >/dev/null 2>&1; then
            echo "(not running)."
            exit 0
         else
            echo "(failed)."
            exit 1
         fi
      fi
      ;;
   restart|force-reload)
      $0 stop
      exec $0 start
      ;;
   *)
      echo "Usage: /etc/init.d/${NAME} {start|stop|restart|force-
      reload}" >&2
      exit 1
      ;;
   esac
   exit 0
```

Ahora inicie saslauthd:

```
# /etc/init.d/saslauthd start
Starting SASL Authentication Daemon: changed ownership of
'/var/spool/postfix/var/run/saslauthd' to root:sasl
saslauthd.
```

Para ver si SMTP-AUTH y TLS funcionan correctamente, ejecute el siguiente comando:

```
# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.
220 server1.centralsoft.org ESMTP Postfix (Debian/GNU)
```

Esto establece una conexión con Postfix. Ahora introduzca:

```
# ehlo localhost
```

Si puede ver estas líneas:

```
server1:/etc/postfix# telnet localhost 25
Trying 127.0.0.1...
```

```
Connected to localhost.localdomain.
Escape character is '^]'.
220 server1.centralsoft.org ESMTP Postfix (Debian/GNU)
ehlo localhost
250-server1.centralsoft.org
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-AUTH LOGIN PLAIN
250-AUTH=LOGIN PLAIN
250 8BITMIME
```

su configuración debería funcionar y ya habrá completado esta parte de la configuración del correo. Puede teclear quit y pasar a la siguiente sección.

Haciendo funcionar Apache

Como mencionamos anteriormente en este mismo capítulo, vamos a incluir un servidor Web en nuestra configuración inicial porque es importante aprender algunos aspectos básicos de administración de servidores, y porque el servidor puede servir de alojamiento para otras herramientas. Al final del capítulo, lo usaremos para ofrecer estadísticas Web generadas con Webalizer.

En noviembre de 2006, Netcraft publicó un informe en el que se decía que el 60 por 100 de los sitios Web en Internet usaban Apache. Esto lo convierte en el servidor más usado de todos los otros servidores juntos.

Apache está bien integrado con la mayoría de las distribuciones Linux. En esta sección, seguiremos un patrón familiar e instalaremos y configuraremos Apache ejecutando el siguiente comando:

```
# apt-get install apache2 apache2-doc
Setting up ssl-cert (1.0-11) ...
Setting up apache2-utils (2.0.54-5) ...
Setting up apache2-common (2.0.54-5) ...
Setting Apache2 to Listen on port 80. If this is not desired, please edit
/etc/apache2/ports.conf as desired. Note that the Port directive no longer
works.
Module userdir installed; run /etc/init.d/apache2 force-reload to enable.
Setting up apache2-mpm-worker (2.0.54-5) ...
Starting web server: Apache2.
Setting up apache2 (2.0.54-5) ...
Setting up apache2-doc (2.0.54-5) ...
```

Una vez que Debian acabe de instalar adecuadamente el servidor apache httpd, ejecute lo siguiente:


```
# apt-get install libapache2-mod-php4 libapache2-mod-perl2 \
php4 php4-cli php4-common php4-curl php4-dev php4-domxml \
php4-gd php4-imap php4-ldap php4-mcal php4-mhash php4-mysql \
php4-odbc php4-pear php4-xslt curl libwww-perl imagemagick
```

Este comando captura y configura 48 archivos, por lo que tardará un rato. Una vez hecho, puede saltar al siguiente paso. Cambie la directiva `DirectoryIndex` del archivo `/etc/apache2/apache2.conf`:

```
DirectoryIndex index.html index.cgi index.pl index.php index.xhtml
```

por:

```
DirectoryIndex index.html index.htm index.shtml index.cgi index.php
index.php3 index.pl index.xhtml
```

Luego, añada símbolos # tal y como se muestra, para comentar las siguientes líneas en el archivo `/etc/mime.types`:

```
#application/x-httpd-php                  phtml pht php
#application/x-httpd-php-source           phps
#application/x-httpd-php3                 php3
#application/x-httpd-php3-preprocessed    php3p
#application/x-httpd-php4                 php4
```

También necesitará comentar dos líneas más en `/etc/apache2/mods-enabled/php4.conf`:

```
<IfModule mod_php4.c>
#AddType application/x-httpd-php .php .phtml .php3
#AddType application/x-httpd-php-source .phps
</IfModule>
```

Después, asegúrese de que las siguientes dos líneas están presentes en el archivo `/etc/apache2/ports.conf`, añádalas si es necesario.

```
Listen 80
Listen 443
```

Ahora, tiene que activar algunos módulos Apache (SSL, rewrite y suexec) para lo que tendrá que crear los siguientes enlaces simbólicos en el subdirectorio mods-enabled:

```
# cd /etc/apache2/mods-enabled
# ln -s /etc/apache2/mods-available/ssl.conf ssl.conf
# ln -s /etc/apache2/mods-available/ssl.load ssl.load
# ln -s /etc/apache2/mods-available/rewrite.load rewrite.load
# ln -s /etc/apache2/mods-available/suexec.load suexec.load
# ln -s /etc/apache2/mods-available/include.load include.load
```

Como vio al instalar otros procesos en secciones previas de este capítulo, instalar los módulos apropiados con apt-get inicia automáticamente Apache en el sistema. Sin embargo, y debido a que ha hecho varios en la configuración, necesitará reiniciar Apache para que los cambios surtan efecto sin tener que reiniciar el servidor. Introduzca este comando:

```
# /etc/init.d/apache2 restart
```

El servidor Web se reiniciará y activará los nuevos módulos, junto con los cambios de configuración.

Añadiendo servicios FTP con ProFTPD

Junto con el servidor httpd que muestra páginas Web en un navegador, necesitará implantar un servidor de trasferencia de archivos (FTP). Usaremos una herramienta de código abierto llamada ProFTPD para este propósito porque es popular, segura y configurable.

El servidor FTP usa un archivo de configuración principal, con directivas y grupos de directivas que cualquier administrador que haya usado alguna vez el servidor Apache comprenderá. ProFTPD tiene archivos `.ftpaccess` de configuración por directorios, de manera análoga a los archivos `.htaccess` de Apache, que obligan a los usuarios a introducir sus ID de usuarios y sus contraseñas para acceder a los directorios individuales.

ProFTPD le permite configurar múltiples servidores FTP virtuales y servicios de FTP anónimos. No invoca ningún programa externo y se ejecuta como usuario sin privilegios. Instale ProFTPD ejecutando este comando:

```
# apt-get install proftpd
```

La figura 2.9 muestra la pantalla que verá una vez que Debian haya descargado y comience a instalar ProFTPD. Éste puede ejecutarse bien como aplicación autónoma o bien como un servicio desde inetd. Por razones de seguridad, ejecutaremos ProFTPD en modo autónomo.

Posteriormente, añada las siguientes líneas al archivo `/etc/proftpd.conf`:

```
DefaultRoot ~
IdentLookups off
ServerIdent on "FTP Server ready."
```

Ahora, al igual que hemos hecho con los otros procesos, reinicie ProFTPD usando este comando:

```
# /etc/init.d/proftpd restart
```

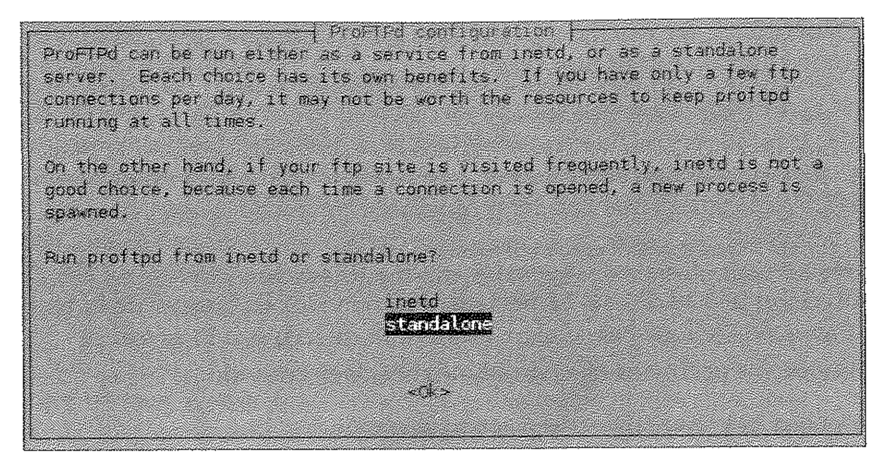



Figura 2.9. Pantalla de configuración Debian para ProFTPD.

Recopilando las estadísticas Web con Webalizer

Webalizer crea informes estadísticos a partir de los archivos del servidor Web. Puede usarlo con un navegador Web estándar y produce informes detallados y fácilmente configurables en formato HTML. El proyecto Debian incluye Webalizer en sus repositorios estables, por lo que puede instalarlo con este comando:

```
# apt-get install webalizer
```

Durante la instalación, necesitará verificar el directorio de instalación (`/var/www/webalizer`), el nombre que se usará para los títulos de los informes estadísticos (podría especificar un nombre de dominio, por ejemplo) y la localización del archivo log del servidor Web (que en su sistema está en `/var/log/apache/access.log.1`):

```
Which directory should webalizer put the output in?
/var/www/webalizer
Enter the title of the reports webalizer will generate.
Usage Statistics for server1.centralsoft.org
What is the filename of the rotated webserver log?
/var/log/apache/access.log.1
```

Sincronizando el reloj del sistema

Los relojes de los ordenadores tienden a desviarse. Por tanto, existe una tarea básica de configuración que conecta su sistema con un servidor *Network Time Protocol* (NTP) que mantendrá la hora correcta.

Para sincronizar su sistema con un servidor NTP, añada las siguientes líneas a /var/spool/cron/crontabs/root:

```
# update time with NTP server
0 3,9,15,21 * * * /usr/sbin/rdate 128.2.136.71 | logger -t NTP
```

Si el archivo no existe, puede crearlo con el comando:

```
# touch /var/spool/cron/crontabs/root
```

La dirección IP 128.2.136.71 pertenece al servidor de tiempo de la Universidad de Carnegie Mellon. Puede usar un tiempo diferente si lo desea.

Modifique los permisos en el archivo crontab ejecutando:

```
# chmod 600 /var/spool/cron/crontabs/root
```

y reinicie el servicio cron mediante:

```
# /etc/init.d/cron restart
```

Instalando los diferentes módulos Perl requeridos por SpamAssassin

Muchas herramientas dependen del lenguaje de programación Perl u ofrecen una interfaz Perl que permite personalizarlas (aunque otros lenguajes están ganando adictos en los mundos del software libre y de Unix). SpamAssassin es una herramienta crítica para administradores de correo (e incluso usuarios de correo), es un programa que usaremos en este libro y está implementando en Perl. Como administrador del sistema, incluso aunque no quiera programar en Perl, debería ser capaz de descargar módulos Perl del repositorio más popular y seguro, el *Comprehensive Perl Archive Network* (CPAN).

Para darle una ligera idea de cómo instalar módulos Perl, añadiremos algunos usando la consola CPAN de Perl. Que es un entorno para buscar archivos e instalar módulos.

Entre en la línea de comandos como root y ejecute el siguiente comando para entrar en la consola CPAN de Perl:

```
server1:/home/admin# perl -MCPAN -e shell
/etc/perl/CPAN/Config.pm initialized.
```

Responda a todas las preguntas presionando la tecla **Intro** para aceptar los valores por defecto. Luego ejecute los siguientes comandos para instalar los módulos que usaremos en el próximo capítulo:


```
> install HTML::Parser
> install DB_File
> install Net::DNS
```

Y con respeto a activar los test, responda no.

Si un módulo ya existiera en el sistema, verá un mensaje como HTML:Parser se ha actualizado. Cuando un módulo se instale con éxito, verá algo como /usr/bin/make install - OK.

Una vez que esté hecho, simplemente pulse q para abandonar Perl y volver a la consola del sistema.

Qué es lo próximo

Ahora que ya ha completado las tareas asociadas a la configuración del servidor, ahora podrá empezar a usarlo en modo de producción. Necesitará configurar los servicios de DNS y notificar al registrador dónde ha configurado su dominio (esto será objeto de estudio en el siguiente capítulo). Una vez que la configuración DNS esté acabada, puede instalar una aplicación basada en Web (nosotros usaremos ISPConfig) y comenzar a explorar cómo funcionan las aplicaciones Web.

Capítulo 3

El sistema de nombres de dominio



Este capítulo le muestra cómo construir un Sistema de Nombres de Dominio (DNS) usando BIND. Cuando acabe este capítulo, debería saber instalar, configurar, mantener y resolver las incidencias de cualquier dominio que registre. Empezaremos con una introducción a DNS, la cual se podrá saltar para ir a la sección de instalación y configuración paso por paso. Si le surgen problemas, siempre puede volver atrás y leer o revisar el material inicial.

Aspectos básicos de DNS

Si busca un poco acerca de DNS en Internet, podrá comprobar que DNS es la base de datos más grande del mundo. Aunque comparándola con otros sistemas gestores de bases de datos como Oracle o MySQL es un poco diferente. De hecho, DNS es el directorio digital distribuido más grande del mundo. Al igual que un directorio telefónico que se usa para asociar nombres con números, DNS asocia la dirección IP con el nombre de los servidores conectados a Internet, que abarcan desde pequeños sitios Web hasta granjas de servidores como Google o Amazon.

Al igual que las bibliotecas públicas tienen una colección de guías telefónicas clasificadas por estados, DNS separa los dominios en categorías. La colección de categorías reside en lo que se llama el directorio raíz. Esta colección está divida en dominios de alto nivel (TLD), de manera similar a como la colección de guías telefónicas está dividida en estado. En lugar de buscar un número de teléfono con el código de área de Nueva York, DNS busca los nombres según el sufijo .edu, .org, .com, .net, .mil, .de, .fr, etc. Los dominios de cada TLD apuntan a la dirección que puede usar para comunicarse con el servidor.

El DNS (que originalmente se definió en el RFC 882 en 1983 y luego se revisó como RFC 1034 y 1035) introdujo varias ideas para gestionar el mapeo de los nombres comunes de Internet a direcciones IPS. El sistema distribuye los datos y los nombres de los equipos de manera jerárquica en un espacio de nombres de dominio. Cada dominio se parece a una rama de un árbol y cada rama contiene a su vez sub-ramas. Los programas llamados servidores de nombres ofrecen información sobre las partes del árbol y otros programas llamados *resolvers* piden información a los servidores de nombres de parte de los programas clientes.

Los esquemas de nombres jerárquicos como DNS evitan la duplicación de los datos. Cada dominio es único, y puede tener tantos servidores como quiera para su dominio, simplemente debe añadir un prefijo a los equipos del dominio. Un sitio que controle `centralsoft.org`, por ejemplo, puede tener equipos que se llamen `server1.centralsoft.org`, `ldap.centralsoft.org` y `mail.centralsoft.org`.

Ventajas de la administración localizada de DNS

Las organizaciones pequeñas a menudo permiten a sus ISP administrar el DNS por ellos. Aunque configurar sus propios servidores tiene ventajas. Le da el control total sobre qué sistemas alojan los servicios públicos (por ejemplo, servicios Web o correo electrónico), y poner DNS en su infraestructura permite mayor escalabilidad: puede añadir servidores según se vaya necesitando e incluso balancear la carga entre ellos. Esto se convierte en importante si posee y opera en varios dominios activos o en servicios internos de autentificación. También tiene más control a la hora de mantener los nombres actualizados. Resumiendo, es valioso controlar su propios DNS en el panorama actual, en lugar de tener a alguien que lo haga.

Muchas empresas han migrado a la Web sus principales procesos de negocio. En lugar de reemplazar los sistemas existentes, prefieren ofrecer sus aplicaciones a través de interfaces Web novedosas. Lo consiguen usando sistema basados en Web capaces de conectar sistemas heterogéneos. Los departamentos de tecnologías de la información usan servidores de aplicaciones como JBoss (propiedad de Red Hat) Webshpere de IBM o BEA WebLogic en segundo plano y otros productos para la presentación. En cada caso, DNS es una parte integrante de los sistemas basados en Web, porque dichos sistemas usan servidores de directorios para comunicarse.

DNS también ocupa un lugar importante en áreas emergentes como la de los servicios Web y la Internet ejecutable, donde la gente puede usar aplicaciones ofrecidas por Google, Yahoo y otros. Resolver las direcciones IP de forma rápida y segura es importante para el éxito de estos productos en Internet y en empresas. Por tanto, considere la configuración y la gestión de DNS como uno de los conocimientos más importantes de la administración de sistemas que puede poseer.

¿Por qué podría necesitar un administrador de sistema gestionar sus propios servidores DNS? Usted debe ofrecer las direcciones de dos o más servidores de su dominio al registrador (por lo menos dos, para garantizar que alguno de los dos funcione cuando alguien solicite un nombre). Debe gestionar los nombres de dominio de los sistemas que van a ser públicos: los servidores Web, los servidores de correo, etc. A medida que aprenda DNS, se irá dando cuenta de que es muy intuitivo. Muchas veces la jerga parece un idioma extranjero. No le encontrará el sentido hasta que no haya trabajado con ella durante un tiempo. Le enseñaremos cómo levantar un servidor DNS en un momento. Luego, revisaremos algunos aspectos y términos clave antes de sumergirnos en los archivos de configuración.

Introducción a BIND

La mayoría de los servidores DNS del mundo se ejecutan gracias al Sistema de Nombres de Dominio de Berkeley o BIND. BIND es un estándar en todas las versiones de Unix y de Linux. Puesto que los administradores necesitan usarlo, este capítulo cubrirá BIND en detalle.

Nota: La alternativa más popular para BIND es la suite djbdns. Funciona bien, la usan muchos servidores de nombres y es fácil de configurar. Véase `http://cr.yp.to/djbdns.html` para más detalles.

No vamos a ofrecer una clase de historia sobre BIND, porque el lector probablemente se dormiría. Sólo señalaremos una anécdota histórica y es que hay gente que todavía usa la antigua y obsoleta versión 4 de BIND. En este capítulo, usaremos la nueva versión 9.

Si usa un sistema con la sintaxis de los archivos de configuración DNS distinta de la que se muestra en este capítulo, probablemente el sistema esté usando BIND 4. Como dijimos antes, las empresas odian reemplazar los sistemas que funcionan, debería ocurrir una catástrofe para que un departamento de tecnologías de la información pudiera actualizar a BIND 8 o 9. Debido a que hay graves riesgos de seguridad para BIND 4, le recomendamos que se actualice (y ya de paso, salte a la versión 8 como mínimo, no a la versión 5, 6 ó 7).

Componentes de BIND

BIND viene con tres componentes. El primero es el servicio o demonio que ejecuta la parte servidora de DNS. Este componente se llama `named`. Es el encargado de responder al teléfono cuando suena.

El segundo elemento de BIND es la librería resolutora. Este componente es el que los navegadores Web, el software de correo y otras aplicaciones consultan cuando intentan encontrar un servidor por su nombre DNS en la jungla de Internet.

Algunas personas piensan que un resolutor es un cliente dentro de BIND. Pero al contrario que el servidor, el cliente no es un programa simple, sino que es una librería que enlaza con cada navegador Web, cliente de correo, etc. El código del resolutor lanza consultas sobre los servidores DNS para intentar traducir nombres en direcciones IP.

Este elemento de BIND usa su propio directorio llamado `resolv.conf` que está presente en cada ordenador. Es su tarea configurar `resolv.conf`. He aquí cómo se vería el archivo `resolv.conf` en los equipos del dominio `centralsoft.org`:

```
search centralsoft.org
nameserver 70.253.158.42
nameserver 70.253.158.45
```

Como puede ver, el archivo de configuración del resolutor BIND es simple. La primera línea busca un servidor en el dominio local. Las otras líneas indican direcciones de servidores de nombres que el administrador conoce, si una consulta falla, se procede con la siguiente línea.

La tercera parte de BIND ofrece herramientas tales como el comando `dig` para probar DNS. Vaya a la consola, teclee dig `yahoo.com` (u otro dominio conocido) y vea lo que ocurre. Analizaremos la herramienta `dig` y otras utilidades del kit más tarde.

Configurando un servidor DNS

Para levantar nuestro servidor, vamos a hacer una instalación de la última versión estable de Debian y configurarla con el número mínimo de paquetes. Si todavía no tiene el disco de instalación basada en red usado en el capítulo anterior, descárguelo de `http://www.us.debian.org/CD/netinst`. Realice una instalación de red y asegúrese de indicar un nombre de dominio adecuado. Luego configure debían como se sugiere aquí.

Cuando tenga la versión actual de Debian GNU/Linux, encontrará diferencias entre ésta y la versión que usamos para escribir las siguientes instrucciones. Los desarrolladores de Linux actualizan sus distribuciones frecuentemente, y los procesos de instalación cambian debido a las actualizaciones, los parches y las nuevas versiones del kernel de Linux. Si encuentra diferencias en los procesos de instalación que describimos, busque la esencia del asunto que explicamos y no tendrá problemas para instalar la última versión.

Después de las etapas iniciales de la instalación de Debian, verá una pantalla gráfica indicándole que escoge el tipo de instalación que desea. La pantalla será más o menos así:

```
( ) Entorno de escritorio
( ) Servidor Web
( ) Servidor de impresión
( ) Servidor DNS
( ) Servidor de archivos
( ) Servidor de correo
( ) Base de datos SQL
( ) Selección manual de paquetes
```

No seleccione ninguna opción, solo pulse la tecla **Tab**. Haga clic en el botón **OK** y el instalador Debian comenzará a descargar y a instalar paquetes. Durante la descarga, verá una pantalla gráfica. Esta pantalla le preguntará si quiere configurar Exim (Exim-config). Elija Sin configuración. Le preguntará "¿Seguro que desea dejar el sistema de correo sin configurar?" Responda **Sí**.

Una vez que se haya completado la instalación mínima de Debian, debería eliminar algunos programas innecesarios que tiene alguna utilidad en una LAN pero no en un servidor de correo de Internet: puede eliminarlos usando la utilidad apt-get:

```
# apt-get remove lpr nfs-common portmap pidentd pcmcia-cs pppoe \
pppoeconf ppp pppconfig
```

Si ha decidido usar SUSE o Fedora en lugar de Debian, puede eliminar estos paquetes con su método preferido.

Ahora, desactivemos algunos servicios y reiniciemos inetd:

```
# update-inetd --remove daytime
# update-inetd --remove telnet
# update-inetd --remove time
# update-inetd --remove finger
# update-inetd --remove talk
# update-inetd --remove ntalk
# update-inetd --remove ftp
# update-inetd --remove discard
# /etc/init.d/inetd reload
```

Para instalar BIND en su servidor Debian, ejecute el comando:

```
# apt-get install bind9
```

Debian descargará el archivo y lo configurará como un servicio de Internet. Podrá ver los siguientes mensajes en la consola:

```
Setting up bind9 (9.2.4-1)
Adding group 'bind' (104)
```

```
Done.
Adding system user 'bind'
Adding new user 'bind' (104) with group 'bind'.
Not creating home directory.
Starting domain name service: named.
```

Usando un entorno chroot seguro

Muchos administradores de seguridad recomiendan ejecutar BIND como un usuario no root en un directorio aislado llamado entorno chroot. Esto protege contra oportunidades de explotar fallos que se detecten en la versión de BIND, que podrían desembocar en que un extraño atacará el demonio named y consiguiera acceso al sistema. Incluso si se ataca named, un entorno chroot limita cualquier daño que pueda hacerse a los servicios de nombres.

Para poner BIND en un entorno chroot, necesita crear un nuevo directorio donde el servicio puede ejecutarse sin estar expuesto a los otros procesos. También se ejecutará como usuario sin privilegios y sólo el root podrá acceder al directorio. El directorio contendrá todos los archivos que BIND necesita, y parecerá ser un sistema de archivos totalmente completo después de ejecutar el comando chroot.

Primero pare el servicio ejecutando este comando:

```
# /etc/init.d/bind9 stop
```

Luego, edite el archivo /etc/default/bind9 y así el demonio se ejecutará como un usuario sin privilegios, defina el entorno chroot como /var/lib/named. Cambie la línea:

```
OPTS="-u bind"
```

para que ponga:

```
OPTIONS="-u bind -t /var/lib/named"
```

Para ofrecer un entorno completo para la ejecución de BIND, cree los directorios necesarios en /var/lib:

```
#mkdir -p /var/lib/named/etc
# mkdir /var/lib/named/dev
# mkdir -p /var/lib/named/var/cache/bind
# mkdir -p /var/lib/named/var/run/bind/run
```

Luego mueva el directorio config desde /etc hasta /var/lib/named/etc:

```
#mv /etc/bind /var/lib/named/etc
```


Ahora cree un enlace simbólico que apunte al nuevo directorio config desde el emplazamiento antiguo y así evitar problemas cuando BIND se actualice en el futuro:

```
#ln -s /var/lib/named/etc/bind /etc/bind
```

Crea dispositivos null y random para que BIND los use y asigne los permisos de directorio:

```
#mknod /var/lib/named/dev/null c 1 3
# mknod /var/lib/named/dev/random c 1 8
```

Luego, cambia los permisos y la propiedad de los archivos:

```
#chmod 666 /var/lib/named/dev/null /var/lib/named/dev/random
# chown -R bind:bind /var/lib/named/var/*
# chown -R bind:bind /var/lib/named/etc/bind
```

También necesita cambiar el script de inicio /etc/init.d/sysklogd para que pueda ver los mensajes en los logs del sistema. Cambie la línea:

```
SYSLOGD=""
```

por:

```
SYSLOGD="-a /var/lib/named/dev/log"
```

Ahora reinicie el proceso de log con el comando:

```
#/etc/init.d/sysklogd restart
```

Verá el siguiente mensaje:

```
Restarting system log daemon: syslogd.
```

Finalmente, inicie BIND:

```
#/etc/init.d/bind9 start
```

Compruebe /var/log/syslog por si hubiera errores. Puede moverse por el contenido del archivo usando:

```
#less /var/log/syslog
```

Normalmente, sabrá si BIND se ha iniciado sin problemas si el archivo syslog muestra:

```
Starting domain name service: named.
```

Desgraciadamente, named puede iniciarse pero fallar al cargar los archivos de datos iniciales, lo que lo dejaría inoperativo. Por tanto, compruebe si named está funcionando introduciendo:

```
#rndc status
number of zones: 6
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
server is up and running
server1:/home/admin#
```

En cambio, si DNS no está funcionando correctamente, podrá ver algo como lo siguiente:

```
#rndc status
rndc: neither /etc/bind/rndc.conf nor /etc/bind/rndc.key was found
```

Si obtuvo este error, échele un vistazo al final de este capítulo.

Configurando un servidor DNS autoritativo

Si quiere encontrar el número de teléfono de Jane Doe en una guía telefónica, la compañía telefónica publica esa información. Pero si quiere encontrar `janedoe.com`, un administrador de sistemas, tiene que haber publicado el nombre de dominio junto con el número (la dirección IP) y haberlos hecho parte de un directorio DNS distribuido. Los administradores hacen esto creando listas que en la jerga de DNS se llaman archivos de zona de llamada.

Una zona contiene la información de un dominio o, si continuamos con la analogía del teléfono, de una familia. Imagine que hay 15 niños viviendo en su casa, y que alguien que está buscando a uno de ellos le llama. Cada niño tiene un teléfono móvil, pero usted no sabe todos los números de memoria. Sin embargo, sí tiene un listado, es decir, un directorio donde puede buscar el número de teléfono móvil del chico que el que llama está intentando localizar.

De manera análoga, usted puede tener 15 servidores en su centro de datos, o 15 sitios Web alojados en su servidor. Para ilustrar esto, digamos que usted administra un servidor que aloja cinco sitios Web diferentes, cada uno con un nombre de dominio completamente diferente. Supongamos que uno es `centralsoft.org` mientras que los otros son `linhelp.com`, `supportcall.org`, `jdshelp.net` y `linuxconf.net`. Todos los propietarios de los sitios Web tienen que pedirle que maneje sus registros DNS. La versatilidad de BIND le permite manejar varios servidores DNS al mismo tiempo, y manejar múltiples dominios independientes alojados en el mismo servidor.

Cada sitio Web es un dominio diferente, por lo que tiene que escribir archivos de zona para cada sitio Web. En las bases de datos de los registrantes, su servidor DNS estará listado como el servidor de nombres de sus nombres de dominio. En otras palabras, `server1.centralsoft.org` está listado como el chico con el que los de fuera tienen que contactar para encontrar a los otros chicos de la casa (`linhelp.com`, `supportcall.org` y los otros).

El archivo que se corresponde con la lista de números de teléfonos móviles de nuestra casa es `/etc/named.conf`. En cierto modo, `/etc/named.conf` es su directorio de archivos de zona, puesto que ofrece información acerca de la localización de cada zona en su sistema.

Su responsabilidad en DNS

Como se comentó anteriormente, DNS distribuye su directorio. Cuando usted paga una tasa y registra un dominio, una de las cuestiones a las que ha de responder tiene que ver con sus servidores de nombres. Usted tiene que ofrecer los nombres y las direcciones de dos servidores, y éstos tienen que estar registrados en el sistema DNS.

Ahora ya se puede hacer una idea de lo que involucra el trabajo de administrador de sistema. Tiene que configurar los servidores de nombres de sus dominios conforme a las especificaciones de la *Internet Engineering Task Force* (IETF). Si no sigue los protocolos especificados, su sistema no pasará a formar parte del servicio de directorio universal. Afortunadamente, lo anterior le ha dado la idea de qué es DNS. Ahora le explicaremos cómo conseguir entrar a formar parte de ese directorio de trabajo.

El método distribuido para resolver nombres de dominio

Revisemos la estructura del directorio DNS de nuevo. El directorio tiene tres niveles. El primer grupo de servidores que se llama servidores raíz, debido a que ofrecen el punto de inicio para las consultas. El segundo grupo que consiste en los servidores de dominios de alto nivel. El TLD incluye .com, .net, .org, .mil, .gov, .edu, etc, además de los dominios de cada país como .de. (Los nombres de dominio no son sensibles a las mayúsculas, es decir .com y .COM son el mismo dominio.)

La figura 3.1 describe la estructura DNS. En la parte de arriba de la figura, puede ver una representación de los servidores raíz de Internet. Estos servidores contienen sólo los nombres y la dirección IP del siguiente nivel de servidores y sólo son responsables de redirigir las peticiones a los TLD concretos.

En el centro de la figura, puede ver algunos de los servidores del `TLD .org`. Estos servidores contienen los nombres y las direcciones IP de todos los servidores DNS registrados con el sufijo .org. Si registra un dominio con el sufijo .org, su dirección IP residirá en cada servidor `TLD .org`. Tendrá que ofrecer la información restante de cada subdominio, incluyendo los servidores de su dominio.

La parte de abajo de la figura 3.1 representa un servidor de nombres primario llamado `server1.centralsoft.org`. Funciona como el servidor de nombres para un número determinado de dominios, como verá más tarde. Por ahora, sólo necesita saber que `server1.centralsoft.org` representa la parte del sistema DNS que tiene que gestionar.

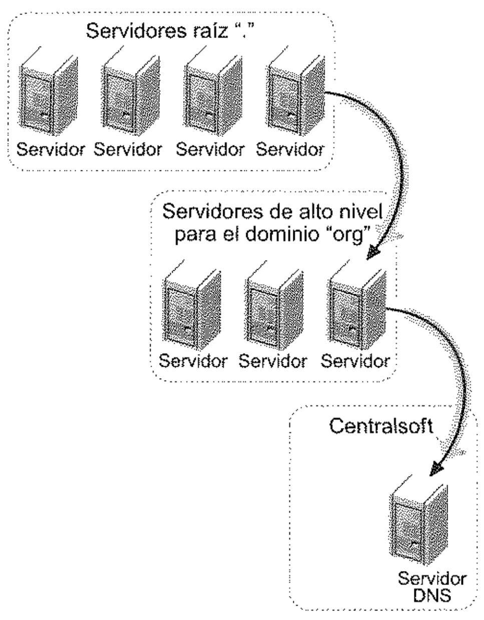


Figura 3.1. Estructura del directorio DNS distribuido.

Encontrando un dominio

Como se mencionó anteriormente, además de proveer un demonio para escribir entradas en el directorio distribuido, BIND ofrece mecanismos para leer el directorio. Cuando su equipo necesita encontrar la dirección de un sitio Web, consulta los servidores DNS que especifique (que normalmente está en su red local o en su ISP).

Imaginemos que su navegador quiere encontrar `www.google.com`. El "cliente" BIND ejecuta un comando que esencialmente pregunta al servidor DNS si

conoce la dirección de sitio Web. Si el servidor DNS no conoce la dirección, pregunta al servidor raíz dicha dirección.

El servidor raíz responde "No la conozco, pero sé dónde puede encontrarla la respuesta. Empiece por los servidores `TLD .com`". Y ofrece la dirección IP de un servidor que conoce los dominios(que son muchos) que están registrados bajo .com. En nombre de tu navegador, el resolutor del servidor DNS luego consulta un servidor .com para la dirección. El servidor .com dice "No tengo la información, pero conozco un servidor de nombres que sí la tiene. Tiene la dirección 64.233.167.99 y su nombre es ns1.google.com".

Su servidor DNS se dirige a la dirección, lee la información que el directorio `ns1.google.com` ofrece y regresa a decirle a su navegador la dirección de `www.google.com`. Luego, el servidor DNS coloca esa información en su caché para no tener que volver a buscar la dirección de Google de nuevo.

Básicamente, `resolv.conf` controla las consultas que los navegadores y otros clientes hacen sobre nombres de dominios, y named responde a las consultas y se asegura que la información se mantenga actualizada para todos los servidores.

Respondiendo consultas

La figura 3.2 describe el proceso para responder a una consulta. Vamos a analizarla. En la esquina superior izquierda de la figura está dibujada la torre de un servidor (en nuestro ejemplo este servidor se llama `server1.centralsoft.org`; realiza la misma función que `ns1.google.com`). Suponemos que el servidor está ejecutando Linux y BIND. Un servidor del nivel más alto dirige a los resolutores al sistema (en el caso de `server1.centralsoft.org`, un servidor de nombres TLD para el dominio `.org` envía las peticiones).

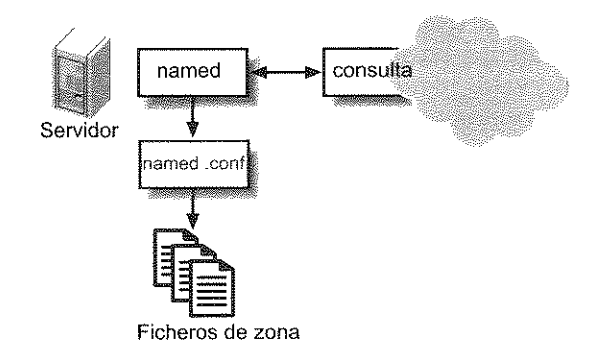


Figura 3.2. Respondiendo a una pregunta.

El demonio escucha en el puerto 53 UDP para cualquiera que haga peticiones de nombres del dominio. Cuando named recibe una petición, consulta su

archivo de configuración, `/etc/named.conf`. Si el servidor tiene información del dominio en cuestión, mira en el archivo de la zona apropiada. Si el archivo de zona tiene la información solicitada, el servidor se la envía al sistema que pidió dicha información.

Algunas personas se refieren a los archivos de configuración como archivos de regla. Esto tiene algo de sentido porque una operación DNS correcta requiere que se cumplan ciertas normas y protocolos. No obstante, los archivos de zona actualmente funcionan como parte del directorio DNS. Su función primaria es ofrecer información, no cumplir las reglas.

Servidores DNS primarios y secundarios

Como dijimos anteriormente, tiene que proporcionar el nombre de al menos dos servidores DNS cuando registre su dominio. Si quiere, puede hacer un duplicado exacto de la información que usó para el primer servidor DNS y colocarlo en el lugar del segundo servidor. Algunos proveedores hacen esto, pero una práctica más común y más útil es considerar a un servidor como el primario o servidor maestro (donde se harán todas las actualizaciones manuales) y otro servidor secundario o servidor esclavo. Luego, BIND permite que el servidor secundario se comunique con el primero y automáticamente replique el directorio, lo que en la práctica se llama zona de transferencia.

Los servidores secundarios son autoritativos, al igual que los servidores primarios. Es decir, los servidores secundarios pueden responden a las consultas y dar información de todas las zonas de las que son responsables. La diferencia es que cuando hace cambios, debería hacerlos solamente en el servidor primario. Los servidores secundarios obtendrán la información del servidor primario.

El servidor primario no transfiere la nueva configuración a los servidores secundarios inmediatamente. En su lugar, cada servidor secundario sondea al servidor primario a intervalos regulares de tiempo para detectar los cambios que se han producido. Un servidor secundario sabe que debería sondear a su "hermano mayor", ya que está etiquetado con el término esclavo (slave) en el archivo `named.conf`, como se muestra aquí:

```
zone "centralsoft.org" {
   type slave;
   file "sec.centralsoft.org";
   masters { 70.253.158.42; };
};
```

No discutiremos sobre la sintaxis y el papel de la entrada por ahora. Las cosas importantes en las que hay que fijarse son el tipo esclavo; línea que define a este servidor como secundario, y la línea donde pone maestro (master), que le dice al servidor dónde conseguir la información. En este ejemplo, el maestro es la dirección IP 70.253.158.42. Esta dirección coincide con la que pusimos en el archivo `resolv.conf` antes. El archivo `resolv.conf` ayuda a un cliente a conectarse a DNS, mientras que la entrada anterior ayuda a un servidor secundario a encontrar el servidor primario.

Cuestiones relativas a los cortafuegos

Si tiene un cortafuegos en su servidor primario, asegúrese de desbloquearlo para el puerto UDP 53. Este puerto se usa para recibir y responder las consultas. Si los servidores secundarios residen en la otra parte de un cortafuegos, también tendrá que desbloquear el puerto TCP 53. Los servidores secundarios usan tanto TCP como UDP para realizar las transferencias de zonas, lo cual es necesario para mantener los servidores actualizados.

Designar el servidor secundario como esclavo lo habilita para que periódicamente compruebe si en el servidor primario se han producido cambios en los archivos del directorio del dominio. El archivo `named.conf` de cada servidor especifica cómo se hace el sondeo y la transferencia de zonas. Los valores actualizados indican al servidor secundario con qué frecuencia debe sondear al servidor maestro. El número de serie es un valor que debe incrementar en el servidor primario cada vez que cambie la información que ofrece. El servidor secundario compara el valor primario con su propio valor para determinar si procede una transferencia de zona. El archivo de configuración primario también especifica el valor de reintentos, que el servidor secundario usará para actualizar los valores en caso de que no pueda conectarse al servidor primario. Esto puede suceder si el servidor maestro o la red fallan. En este caso, el servidor secundario enmascara al primario durante ese rato.

Aunque un servidor secundario no puede actuar como mascara indefinidamente. De manera eventual, su información podría caducar, por lo que sería preferible dejar de responder a las peticiones. Por lo tanto, el archivo de configuración también especifica un tiempo de expiración. Si este tiempo transcurre sin que se produzca una actualización, el servidor secundario sigue intentando contactar con el servidor primario pero deja de responder a las peticiones.

Hay algunos valores más que debería conocer antes de manejar los archivos de configuración: el tiempo de vida (TTL). Cuando un servidor DNS remoto recibe una respuesta a una pregunta suya, almacena el caché esa información y la reutiliza mientras sea válido el valor TTL. Esta técnica mejora el rendimiento de DNS. Gracias a la caché, si alguien pasa una hora visitando varias páginas Web de su sitio (cada una de las cuales involucra varias descargas), un servidor próximo al usuario sólo necesitará preguntar el nombre de dominio una vez, puesto que será capaz de satisfacer cada petición en base a la caché. Para evitar que la

información de la caché caduque, TTL asegura que el servidor descartará el valor que le pide, y acudirá al servidor autoritativo para obtener el valor actual.

Verá todos estos valores en su archivo de zona, no en el archivo `named.conf`. El archivo `named.conf` apunta a la localización de su archivo de zona.

Servidores de solo caché

Además de los servidores primario y secundario, DNS ofrece servidores de sólo caché. Los administradores los usan para reducir la carga de los servidores autoritativos. Un servidor de caché no tiene autoridad, simplemente hace que DNS trabaje más rápido almacenando los nombres de dominio que obtiene de los servidores autoritativos y ofreciéndoselos a los clientes.

El servidor que configure para alojar los dominios, a menudo está ocupado respondiendo a las consultas de otros servidores DNS de Internet. Esta tarea consume muchos recursos, por lo que los administradores usan servidores de caché para almacenar localmente la información que los usuarios demandan. Verá servidores de caché usados por los ISP, por ejemplo, para dar servicio a sus clientes. Luego, usan otro servidor para ofrecer nombres de dominio de Internet a los sitios que alojan.

Cuando instala BIND, se configura un servidor de caché por defecto. Cuando realiza una consulta, el servidor de caché mantiene los resultados en la caché. La próxima vez que intente encontrar el mismo sitio Web, no tendrá que repetir el proceso entero: sólo tendrá que obtener la dirección IP del equipo de la caché.

Editando los archivos de configuración

Hasta aquí, hemos hecho una exploración de alto nivel del sistema de nombres de dominio y hemos explicado las partes que tiene que mantener. Ahora vamos a entrar en detalle acerca de los archivos de configuración que puede escribir, modificar o arreglar cuando sea necesario.

Cuando instala BIND en Linux, el paquete genera los archivos de configuración por usted; no tiene que escribir cada archivo desde cero. La figura 3.3 ilustra los archivos básicos. Empezaremos por el archivo `named.conf`, que coordina el sistema entero en cada servidor BIND y apunta al resto.

named.conf

Recuerde que cuando named recibía una petición, consultaba su propio directorio, el archivo de configuración `named.conf`. Esto hace apuntar a named al archivo de zona para el dominio solicitado.

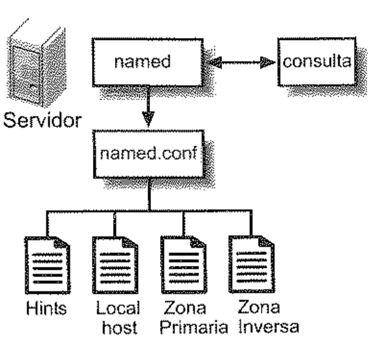


Figura 3.3. Archivos de configuración de BIND.

Veamos atentamente un archivo `named.conf` simple. Si no entiende este punto, sólo trate de familiarizarse con él. Entraremos en más detalles dentro de un momento.

Recuerde, este archivo está normalmente instalado en su servidor Linux por defecto. Dependiendo de la distribución, `named.conf` puede residir en diferentes directorios (está en `/etc/bind/named.conf` para BIND 9 bajo Debian). Su apariencia puede variar ligeramente. Algunas veces, por ejemplo, el archivo viene muy comentado. Aquí tenemos nuestro ejemplo. Los comentarios van a continuación de la doble barra invertida.

```
options {
   pid-file "/var/run/bind/run/named.pid";
   directory "/etc/bind";
   // query-source address * port 53; };
//
// configuración de servidor de nombres maestro
//
zone "." {
   type hint;
   file "db.root";
};
   zone "0.0.127.in-addr.arpa" {
   type master;
   file "db.local";
};
zone "158.253.70.in-addr.arpa" {
   type master;
   file "pri.158.253.70.in-addr.arpa";
};
zone "centralsoft.org" {
   type master;
   file "pri.centralsoft.org";
};
```

Seguridad básica en transferencia de datos

En nuestra configuración actual, cada servidor de nombres tiene permitido transferir nuestra zona `centralsoft.org` desde el servidor de nombres primario. Debido a que queremos permitir sólo a nuestro servidor secundario (70.253.158.45) transferir la zona, debemos añadir la siguiente línea a la zona `centralsoft.org` en el archivo `named.conf` de nuestro servidor de nombres primario, `server1.centralsoft.org`:

```
allow-transfer { 70.253.158.45; };
```

La zona debería quedar así:

```
zone "centralsoft.org" {
   type master;
   file "pri.centralsoft.org";
   allow-transfer { 70.253.158.45; };
};
```

El archivo de ejemplo `named.conf` alude a los otros cuatro archivos de configuración. La tercera línea tiene el directorio que los contiene, `/etc/bind`.

La sentencia de opciones contiene dos líneas. La primera muestra la localización de `named.pid` que simplemente contiene el ID del proceso del demonio named que se está ejecutando. Que puede parecer información extraña para almacenar, pero que es muy útil cuando se quiere parar o reiniciar named. La segunda línea de la sentencia de opciones define el directorio que contiene los archivos relacionados con su ejecución.

Las subsiguientes sentencias de zona, un ejemplo de lo que vimos antes, identifican la localización de varios archivos que contienen la información de configuración. En resumen, `named.conf` necesitará apuntar a los siguientes archivos en las sentencias de zona:

- **Archivo Hints (para la zona "."):** El archivo contiene los nombres y las direcciones de los servidores raíz de Internet. named debe conocer las direcciones de estos servidores para poder empezar a consultar cuando ninguno de los componentes del dominio solicitado esté almacenado en la caché.
- **Archivo local host (para la zona "0.0.127.in-addr.arpa"):** El archivo representa su propio sistema (dirección IP 127.0.0.1). La ventaja de crear archivos de zona local para cada aspecto de su sistema es la de reducir el tráfico y permitir al software funcionar de la misma forma, independientemente de que esté accediendo a un equipo local o a uno remoto.
- **Archivo de zona inversa (para la zona "158.253.70.in-addr.arpa"):** El archivo convierte las direcciones IP en nombres. Es una imagen especular del archivo de zona primaria. Puede reconocer un archivo de zona inversa porque tiene una extensión in-addr-arpa y usa registros PTR (que se describirán después).
- **Archivo de zona primaria (para la zona "centralsoft.org"):** Este archivo, algunas veces llamado base de datos de dominios, define la mayor parte de la información necesaria para resolver las consultas sobre el dominio que administra. No viene preconfigurado cuando instala BIND. Normalmente, tiene que escribir este archivo desde cero o usar uno de los archivos que acompañan a BIND como plantilla. El archivo de zona primaria asocia nombres a direcciones IP y ofrece información sobre los servicios que los equipos ofrecen en Internet (incluyendo servidores Web y FTP, servidores de correo, servidores de nombres, etc.).

El archivo de configuración por defecto contiene las dos primeras sentencias de zona (para los servidores raíz y los archivos locales, estos archivos típicamente aparecen cuando instala BIND y no necesita cambiarlo). Tendrá que añadir entradas para los archivos de zona inversa y zona primaria. Los archivos de zona usan varios tipos de registros, entre los que se incluyen:

- SOA (Comienzo de autoridad).
- NS (Servidor de nombres).
- MX (Pasarela de correo, que identifica un servidor de correo en el dominio).
- A (Asocia un nombre de equipo a una dirección).
- CNAME (Nombre canónico, que define un alias para un equipo en un registro A).
- PTR (Puntero, que asocia direcciones con nombres).

No es necesario intentar memorizar o comprender estos tipos de registros en este punto. Tendrá la oportunidad de usarlos cuando profundicemos en el tema. Ahora, revisaremos en detalle un archivo de zona primaria.

El archivo de zona primaria

El archivo de zona primaria contiene el grueso de la información de configuración que DNS necesita. El formato del archivo no está estandarizado, pero los elementos que contiene están especificados en el RFC 1035.

Si está usando el conjunto de archivos que la instalación ofrece, debería darle un nombre a su archivo de zona primaria añadiendo un prefijo a su dominio. Hemos nombrado el archivo de zona para el dominio `centralsoft.otg` como `pri.centralsoft.org` (el prefijo pri le ayudará a reconocer que es el prima-

rio). Describiremos cada parte del archivo aquí, para verlo entero, eche un vistazo al final del capítulo.

Las primeras líneas ofrecen la información necesaria para sincronizarlo con el servidor secundario o esclavo:

```
@ IN SOA server1.centralsoft.org. root.localhost. (
      2006012103; serial-no
      28800; refresh, seconds
      7200; retry, seconds
      604800; expiry, seconds
      86400 ); minimum-TTL, seconds
;
```

Eso es un registro SOA. SOA es el Comienzo de Autoridad, que lo distingue como información para servidores autoritativos (tanto primarios como secundarios) en contraposición con los servidores de sólo caché. A la vez que escribe su parte en el directorio DNS distribuido, el sistema le cede la autoridad de su parte a usted. Por lo que su archivo de zona tiene que indicar dónde empieza la autoridad, es decir, el dominio que está sirviendo.

Nota: Un punto y coma (;) no indica el final de una línea, sólo indica el comienzo de un comentario. Por tanto, si no quisiera añadir el comentario "serial-no", podría escribir la siguiente línea:

```
2006012103; serial-no
```

simplemente como:

```
2006012103
```

Fijémonos en la primera línea, comienza por el símbolo (@). De izquierda a derecha los campos son:

- Name: El nombre raíz de la zona. El símbolo @ es la referencia abreviada a la zona actual del archivo `/etc/named.conf`. En otras palabras, es equivalente a usar `server1. centralsoft.org` en nuestro ejemplo. El símbolo @ también se conoce como origen en la jerga DNS.
- Class: La clase DNS. Existe un número de clases, pero la gran mayoría de los sitios usan la clase IN (Internet). Las otras clases están para protocolos y funciones que no son de Internet.
- Type: El tipo del registro de recurso DNS. En este caso, es un registro de recurso SOA.
- Nameserver: El nombre completo del servidor de nombres primario. Algunos detalles son bastante importantes: el nombre debe terminar con un punto (.) que denote la raíz de la jerarquía DNS para poder indicar que la ruta es un nombre de dominio completo.
- Dirección de Email: La dirección de correo electrónico de la persona que es responsable del dominio. Hay otra convención DNS específica aquí, no puede usar la @ que aparece en todas las direcciones de correo de Internet, porque como hemos visto, una @ tiene otro significado en este archivo. Por tanto, se sustituye con un punto. Aquí tenemos que especificar el usuario root del sistema local o `root@localhost`, pero tenemos que especificar dicha dirección con un formato inusual: `root.localhost.` Fíjese que la dirección también debe acabar con un punto.

Las siguientes líneas de los registros SOA contienen aspectos necesarios para los servidores esclavos:

- Serial-no: El número de serie para la configuración actual. Usted incrementa este número cada vez que hace un cambio en la configuración, por lo que servidores esclavos sabrán cuándo ha actualizado la información. Este número está a menudo en formato de fecha YYYYMMDD, con una etiqueta de doble dígito al final (lo que le permite editar el archivo varias veces cada día). De esta forma, cada número de serie es mayor que el anterior e indica la fecha en que fueron hechos los cambios. Cada esclavo comprueba periódicamente el número de serie para ver si ha cambiado. Si el número actual del servidor es mayor que el que tiene el esclavo, éste realiza una transferencia de zona. 2006l2103 es el número de serie inicial en nuestro archivo de zona de ejemplo.
- Refresh: El intervalo en el que un servidor DNS esclavo debería sondear al maestro para determinar si es necesaria una transferencia de zona. El valor está representado en segundo. En nuestro archivo de ejemplo, usamos el valor 28800 (28.800 segundos = 8 horas).
- Retry: Con qué frecuencia debería intentar conectarse al maestro si una conexión fallara. El intervalo en nuestro ejemplo es 7200 (7200 segundos = 2 horas).
- Expiry: La cantidad de tiempo que un esclavo debería intentar conectar con el maestro antes de que los datos que contiene caduquen. Si los datos expiran y el esclavo es incapaz de conectar con el servidor para actualizar la información, las futuras consultas irán dirigidas a los servidores raíz. El tiempo especificado aquí es también la cantidad de tiempo que un servidor esclavo debería continuar respondiendo a las peticiones, incluso aunque no haya podido actualizar el archivo de zona; representa el valor que puede tolerar el manejo de información obsoleta. En nuestro ejemplo, usamos 604800 (604.000 segundos = 7 días).

- Minimum-TTL: El tiempo de vida por defecto para este dominio en segundos. Cualquier registro de recursos que no tenga especificado el TTL usa el valor por defecto que es 86400. Ya que 86400 segundos son un día, el registro de consultas a caché durará un día.

El registro SOA va seguido de una lista de nombres de equipos de varios tipos:

```
NS server1.centralsoft.org.;
NS server2.centralsoft.org.;
```

Estos registros NS especifican los servidores de nombres del dominio (los que proporcionó cuando registró el dominio). Una vez más, no es necesario un punto y coma, pero es conveniente en caso de que quiera añadir un comentario al final de la línea.

Lo siguiente es un registro MX que identifica el servidor de correo del dominio:

```
MX 10 server1.centralsoft.org.
```

Hemos usado solo un servidor de correo en nuestro ejemplo, pero la mayoría de entornos de producción ofrecen varios (bien para gran volumen de tráfico o como respaldo por si uno falla). El segundo campo de este registro (10 en nuestro ejemplo) puede usarse para indicar el orden en que debería intentar conectarse con los servidores MX, es decir, prioriza los servidores.

El registro MX de nuestro archivo de zona primaria va seguido por varios registros A:

```
centralsoft.org. A 70.253.158.42
www A 70.253.158.42
server1 A 70.253.158.42
server2 A 70.253.158.45
```

Un registro A asocia un nombre a una dirección IP. Debido a los múltiples nombres que pueden asignarse a un equipo, puede tener múltiples registros A apuntando a una única dirección IP. Sin embargo, cada nombre de equipo no puede tener más de un registro A. Nuestro archivo tiene cuatro registros A, asociando tres nombres a una dirección y otro nombre a otra dirección diferente.

Mejoras y características avanzadas

Si define un archivo con los contenidos de la sección previa, asegúrese de insertar los nombres de equipo apropiados a la direcciones IP de su entorno, tendrá funcionando un archivo de zona primaria (aunque necesitará los otros archivos también, como explicaremos más tarde). No obstante, deberá tener en cuenta algunas cosas útiles que puede hacer con el archivo de zona primaria.

Registros MX

Como ha visto, un registro MX se parece a esto:

```
MX 10 server1.centralsoft.org.
```

Este registro indica la dirección de correo electrónico que el dominio `centralsoft.org` debería despachar al `server1.centralsoft.org` (el servidor de correo para el dominio) cuya prioridad debe ser 10.

Las prioridades entran en juego en configuraciones más complejas, donde más de un servidor de correo está disponible. Números más bajos indican prioridades más altas (el 1 es la prioridad más alta). El sistema de prioridades funciona de esta forma: el servidor de correo remoto intenta conectar con el servidor que en su lista tiene la prioridad más alta, si no responde, el servidor lo vuelve a intentar con el servidor cuya prioridad es la segunda más alta, y así sucesivamente. Por tanto, proporcione una lista con más de una pasarela de correo, como se muestra aquí:

```
MX 10 server1.centralsoft.org.
MX 20 mail.someotherdomain.com.
```

Ahora si el correo va a `centralsoft.org`, el MTA de origen primero intenta conectarse a `server1.centralsoft.org`, ya que tiene la prioridad más alta (10). Si `server1.centralsoft.org` no está disponible, el MTA de origen usará el siguiente servidor, `mail.someotherdomain.com`, que tiene una prioridad de 20.

Nota: DNS no especifica cómo tratar múltiples servidores de correo con la misma prioridad. Muchas pasarelas eligen uno de manera aleatoria para implementar un tipo primitivo de balanceo de carga.

Hasta ahora, hemos definido registros MX sólo para el correo dirigido a `user@centralsoft.org`. ¿Qué sucede si quiere enrutar correo a distintos departamentos de una compañía o a la sección de una agencia gubernamental? Puede hacerlo añadiendo subdominios a sus registros MX.

De esta forma, añadir `accounting.centralsoft.org` simplemente requeriría otro registro:

```
accounting.centralsoft.org.  MX 10 server1.centralsoft.org.
```

Fíjese en el "." al final de `accounting.centralsoft.org`. Si no añade el punto, el origen de la zona se agrega al nombre. Por ejemplo, si escribió:

```
accounting.centralsoft.org  MX 10 server1.centralsoft.org.
```

sin el punto al final, esto se transformaría en `accounting.centralsoft.org. centralsoft.org`, lo que es incorrecto.

Registros A

Los registros NS y MX usan nombres de equipo como `centralsoft.org`, `server1.centralsoft.`

`Org` y `server2.centralsoft.org`, pero el archivo de zona primaria también debe especificar la dirección IP con la que se deben asociar estos nombres. Los registros A realizan esta asociación. Muchas personas los consideran los registros DNS más importantes porque puede usarlos para crear direcciones como `www.centralsoft.org`, donde www es el equipo.

El siguiente registro A simple de nuestro archivo de zona primaria indica que `centralsoft.org` tiene la dirección IP 70.253.158.42:

```
centralsoft.org.        A 70.253.158.42
```

(Recuerde añadir el punto al final del nombre.)

En un navegador, probablemente estará acostumbrado a introducir `www.centralsoft.org` en lugar de `centralsoft.org`. `www.centralsoft.org`, que es técnicamente diferente de `centralsoft.org`, pero la mayoría de los visitantes esperan ver el mismo sitio Web, independientemente de si están incluyendo www. o no. Por tanto, hemos creado este registro:

```
www                     A 70.253.158.42
```

Las www no van seguidas de un punto, por lo que BIND añade el origen de la zona. El efecto es el mismo que especificar:

```
www.centralsoft.org.    A 70.253.158.42
```

Especifique la dirección IP de `server1.centralsoft.org` y `server2.centralsoft.org`:

```
server1                 A 70.253.158.42
server2                 A 70.253.158.45
```

El registro para `server2.centralsoft.org` apunta a diferentes direcciones IP, lo que tiene sentido ya que es nuestro servidor de nombres secundario y por tanto tiene que estar en un sistema diferente para el caso de que nuestro servidor de nombres primario se caiga.

El problema del Bootstrapping y los registros pegamento

Podría preguntarse cómo `server1.centralsoft.org` y `server2.centralsoft.org` pueden estar acostumbrados a buscar registros de `centralsoft.org` si están en la zona que debe buscarse. Este es el clásico problema del bootstraping: no puede usar la misma técnica para iniciar la búsqueda que la que usa para hacer las búsquedas. La solución requiere registros pegamento. Cuando el servidor TLD para .org dirige los sitios remotos a los servidores de nombres de `centralsoft.org`, normalmente les proporciona un nombre en lugar de una dirección IP (por ejemplo, `server1.centralsoft.org` en lugar de 70.253.158.42). Pero para los servidores DNS autoritativos de la zona que se está buscando (en nuestro caso, asociando `server1.centralsoft.org` a 70.253.158.42), los servidores TLD proporcionan la dirección IP en lugar del nombre del servidor de nombres. Esto significa que no tiene que encontrarlo, sino solo preguntar dónde está.

Registros CNAME

CNAME es la abreviatura de nombre canónico; puede pensar que es un alias para un registro A. Por ejemplo:

```
ftp       CNAME www
```

quiere decir que `ftp.centralsoft.org` es un nombre alternativa para `www.centralsoft.org`, por lo que `ftp.centralsoft.org` apunta a la misma máquina que `www.centralsoft.org`. Podría encontrar situaciones, especialmente al descargar paquetes Linux, donde el repositorio fuera `http://ftp.mirrors.kernerl.org`. En estos casos es casi siempre cierto que un registro CNAME se ha usado para asignar la parte ftp del nombre del equipo a un sistema que tiene un nombre distinto en su registro A. Un CNAME debe siempre apuntar a un registro A, no a otro registro CNAME. Además, no debería usar nombres de equipo CNAME en registros MX o SOA. Esto, por ejemplo, no está permitido:

```
MX 10 ftp
```

El uso de registros CNAME tiene pros y contras. Muchos especialistas en DNS desaconsejan su uso. Aunque todavía se puede encontrar funciones para los registros CNAME. Por ejemplo, si su directorio DNS contiene muchos registros A apuntando a la misma dirección IP y migra a otro servicio de alojamiento que asigna una dirección IP distinta, tendrá que actualizar cada registro A. Pero si tiene un solo registro A y todos los otros nombres en registros CNAME, solamente tendrá que actualizar el registro A. Por lo que los registros CNAME todavía juegan un pequeño papel en el mundo del DNS.

Registros TXT y SPF

Los registros TXT le permiten añadir texto a una zona. La gente suele usar los registros TXT para embeber registros SPF (Framework de política de envíos), que controla si las pasarelas de correo deberían aceptar correo enviado a nuestro do-

minio. Los grandes proveedores de correo como Yahoo! y Hotmail usan los registros SPF para evitar el spam que llega a su dominio. Si un correo electrónico llega procedente de un equipo que no está listado en el registro SPF, un MTA lo clasificará como spam. Puede encontrarse un asistente para crear registros SPF en `http://www.openspf.org/wizard.html?mydomain=&x=26&y=8`. Nosotros hemos usado este asistente para crear dos registros SPF para `centralsoft.org`, luego los hemos embebido en registros TXT y después los hemos añadido a nuestro archivo de zona:

```
centralsoft.org.              TXT "v=spf1 a mx ~all"
server1.centralsoft.org.      TXT "v=spf1 a -all"
```

Juntándolo todo

Ahora echemos un vistazo a nuestro archivo de zona, `pri.centralsoft.org`. Fíjese en que hemos añadido registros CNAME y TXT a las partes que comentamos antes:

```
@ IN SOA server1.centralsoft.org. root.localhost. (
                    2006012103; serial-no
                    28800; refresh, seconds
                    7200; retry, seconds
                    604800; expiry, seconds
                    86400 ); minimum-TTL, seconds
;
                 NS server1.centralsoft.org.;
                 NS server2.centralsoft.org.;
;
                 MX 10 server1.centralsoft.org.
;
centralsoft.org. A 70.253.158.42
www              A 70.253.158.42
server1          A 70.253.158.42
server2          A 70.253.158.45
ftp              CNAME www
centralsoft.org.              TXT "v=spf1 a mx ~all"
server1.centralsoft.org.      TXT "v=spf1 a -all"
```

El archivo de zona inversa

Con nuestro archivo de zona primaria completo, los programas pueden buscar el dominio `centralsoft.org` y todos los subdominios DNS. Pero todavía necesitamos un archivo de zona inversa. Un archivo de zona inversa asocia direcciones IP a nombres. Es como si fuera un espejo del archivo de zona primaria, en lugar de listar los nombres primeros, el archivo de zona inversa lista las direcciones IP primero.

¿Por qué querría alguien usar el archivo de zona inversa? En el pasado, muchas organizaciones no le permitían usar sus servicios si no podían hacer un ping inverso a su dominio. Hoy en día, muchos servidores de Internet usan el ping inverso para verificar el origen del correo y así evitar el spam; este es el propósito de los registros SPF que se vieron antes. El sistema que hemos descrito aquí tiene un problema a la hora de despachar correo que será explicado más adelante. El DNS indica qué MTA es responsable del correo para el dominio de la dirección de correo del remitente. Muchos difusores de spam intentan difundir el correo usando distintos MTA, pero el agente de correo receptor no puede hacer una búsqueda inversa, por lo que detecta una irregularidad y rechaza el correo no deseado. Ya que no queremos que el correo originado en el domino `centralsoft.org` sea clasificado como spam, crearemos un archivo de zona inversa. Primero, para apuntar a este archivo, tenemos que colocar esta entrada en nuestro archivo `named.conf`:

```
zone "158.253.70.in-addr.arpa" {
   type master;
   file "pri.158.253.70.in-addr.arpa";
};
```

Los números pueden parecer extraños, pero siguen un patrón simple. `central-soft.org` está en la red 70.253.158, por lo que invertimos los elementos 70.253.158 para producir 158.253.70 y usarlos en la sentencia zona en `named.conf`. El dominio `in-addr.arpa` es el dominio de alto nivel usado para las búsquedas inversas.

Llamaremos a nuestro archivo de zona inversa `pri.158.253.70.in-addr.arpa` y colocaremos el archivo en el mismo directorio que nuestro archivo de zona primaria, `pri.centralsoft.org`. El comienzo de `pri.158.253.70.in-addr.arp` se parece mucho a `pri.centralsoft.org`:

```
@ IN SOA server1.centralsoft.org. root.localhost. (
                    2006012103; serial-no
                    28800; refresh, seconds
                    7200; retry, seconds
                    604800; expiry, seconds
                    86400 ); minimum-TTL, seconds
;
                 NS server1.centralsoft.org.;
                 NS server2.centralsoft.org.;
```

Pero aquí no añadimos ningún registro A, MX o CNAME. En su lugar, creamos registros PTR.

Registros PTR

PTR es la abreviatura de puntero, y es precisamente esto: un puntero a un nombre de dominio. Creemos uno que comience por la dirección de

centralsoft. org, 70.253.158.42. El archivo named.conf ya ha indicado, a través de la sentencia zona que mostramos en la sección anterior, que este archivo define los equipos del dominio 70.253.158. Por lo que todos los registros PTR tienen que especificar la parte final de la dirección IP del equipo, 42:

```
42              PTR    centralsoft.org.
```

Cree exactamente un registro PTR para cada dirección IP de su dominio. Para nuestro ejemplo, la otra dirección IP que usamos es 70.235.158.45 (para server2. centralsoft.org), por lo que añadiremos:

```
45              PTR    server2.centralsoft.org.
```

Entonces esto es todo. Nuestro archivo de zona inversa se debería parecer a lo siguiente:

```
@ IN SOA server1.centralsoft.org. root.localhost. (
                    2006012103; serial-no
                    28800; refresh, seconds
                    7200; retry, seconds
                    604800; expiry, seconds
                    86400 ); minimum-TTL, seconds
;
                 NS server1.centralsoft.org.;
                 NS server2.centralsoft.org.;

42               PTR   centralsoft.org.
45               PTR   server2.centralsoft.org.
```

Búsquedas de prueba

Una vez que ha editado todos los archivos de configuración y de zona, necesita que BIND conozca los cambios. Puede parar y arrancar named así:

```
# /etc/init.d/bind9 stop
# /etc/init.d/bind9 start
```

Si se produce algún error o si el servicio BIND no hace lo que se espera, por favor, diríjase a la sección de resolución de problemas para ver en detalle los problemas más comunes.

En el futuro, si el único cambio que hace es actualizar el archivo de zona con una nueva entrada DNS para el dominio correspondiente, es suficiente con que le indique a BIND que recargue la información de zona (en lugar de reiniciar el servicio entero):

```
#rndc reload centralsoft.org
```


El comando rndc se discutirá con más detalle pronto.

Ahora podemos probar nuestra configuración haciendo una búsqueda con la herramienta de línea de comandos dig. Primero, buscaremos la dirección IP de centralsoft.org:

```
#dig centralsoft.org

; <<>> DiG 9.2.1 <<>> centralsoft.org
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48489
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;centralsoft.org.                IN     A

;; ANSWER SECTION:
centralsoft.org.    86400        IN     A      70.253.158.42
;; Query time: 198 msec
;; SERVER: 81.169.163.104#53(81.169.163.104)
;; WHEN: Sat Mar 11 18:55:21 2006
;; MSG SIZE rcvd: 49
```

Como puede observar, esta búsqueda devuelve automáticamente la dirección IP 70.253.158.42.

Ahora podemos hacer una búsqueda inversa:

```
#  dig -x 70.253.158.42

; <<>> DiG 9.2.1 <<>> -x 70.253.158.42
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4096
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;;42.158.253.70.in-addr.arpa.     IN     PTR

;; ANSWER SECTION:
42.158.253.70.in-addr.arpa. 5304  IN     PTR    centralsoft.org.

;; Query time: 2 msec
;; SERVER: 81.169.163.104#53(81.169.163.104)
;; WHEN: Sat Mar 11 18:57:54 2006
;; MSG SIZE rcvd: 98
```

La búsqueda directa y la inversa se complementan. Nuestro servidor primario está completo.

Configurando el servidor de nombres secundario

Ahora, levantaremos nuestro servidor de nombres secundario, `server2. centralsoft.org`. Actuará como copia de seguridad en caso de que el servidor primario (`server1.centralsoft.org`) falle, por lo que las personas todavía podrán buscar `centralsoft.org` y sus subdominios.

El archivo `named.conf` para `server2.centralsoft.org` es parecido al del servidor de nombres primario con algunas diferencias:

```
options {
   pid-file "/var/run/bind/run/named.pid";
   directory "/etc/bind";
// query-source address * port 53;
};

zone "." {
   type hint;
   file "db.root";
};

zone "0.0.127.in-addr.arpa" {
   type master;
   file "db.local";
};

zone "centralsoft.org" {
   type slave;
   file "sec.centralsoft.org";
   masters { 70.253.158.42; };
};
```

La diferencia más importante es la que se comentó antes en este mismo capítulo. El tipo esclavo, que se indica en la sentencia final indica que es una zona esclava. En la línea del archivo especificamos el nombre del archivo donde se debe guardar la zona esclava, y en la línea maestra especificamos la dirección IP del servidor de nombres primario.

Esto es todo lo que tenemos que hacer para configurar el servidor de nombres secundario.

Reinicie named en `server2.centralsoft.org` y luego debería encontrar el archivo `/etc/bind/sec.centralsoft.org` en su servidor de nombres secundario. ¿Qué ha ocurrido? El servidor de nombres secundario ha contactado con el servidor de nombres primario que le ha transferido la zona.

Ahora, cada vez que actualice una zona en el servidor de nombres primario, asegúrese de que el número de serie se incrementa. En caso contrario, la zona actualizada no se transferirá al servidor de nombres secundario.

Herramientas BIND

Como hemos mencionado anteriormente en este capítulo, BIND se divide en tres partes: el demonio named, la librería resoluta y algunas herramientas.

Una herramienta que ya ha usado es dig, que los administradores usan para consultar los servidores de nombres DNS. dig hace búsquedas DNS y muestra las respuestas devueltas por los servidores de nombres y las estadísticas sobre la consulta.

La mayoría de los administradores DNS usan dig para solucionar los problemas de DNS debido a su flexibilidad, facilidad de uso y claridad. Otras herramientas de búsqueda suelen tener menos funcionalidad. Otra alternativa podría ser, no obstante, nslookup. También echaremos un vistazo a rndc, una herramienta de administración útil que se incluye con BIND.

nslookup

nslookup trabaja de manera similar a dig pero está obsoleto en Linux. Usarlo requiere más trabajo, pero debería serle familiar porque Microsoft Windows aún lo usa como herramienta primaria de búsqueda.

nslookup consulta servidores de nombre de dominio de Internet en dos modos: interactivo y no interactivo. El modo interactivo permite consultar los servidores de nombres para obtener información sobre varios equipos y dominios, o para imprimir una lista de equipos en un dominio. El modo no interactivo simplemente imprime el nombre y la información solicitada para un equipo o dominio. Por ejemplo, podría ejecutar la siguiente búsqueda para encontrar información sobre el servidor Google:

```
#nslookup ns1.google.com
Server:         68.94.156.1
Address:        68.94.156.1#53

Non-authoritative answer:
Name:    ns1.google.com
Address: 216.239.32.10
```

En el modo interactivo, nslookup ofrece una interfaz donde se pueden ejecutar comandos. Por ejemplo:

```
#nslookup
>
```

Desde la interfaz se pueden hacer varias búsquedas simples, como la de una dirección IP:

```
> 70.253.158.42
Server:     172.30.1.2
Address:    172.30.1.2#53

Non-authoritative answer:
42.158.253.70.in-addr.arpa    name = adsl-70-253-158-42.dsl.rcsntx.
                              swbell.net.

Authoritative answers can be found from:
158.253.70.in-addr.arpa nameserver = ns1.swbell.net.
158.253.70.in-addr.arpa nameserver = ns2.swbell.net.
>
```

Puede ejecutar varios comandos, incluyendo lserver (que usa su servidor local para hacer una búsqueda), server (que usa otro servidor para hacer una búsqueda) y host. El comando lserver produce una salida como la siguiente:

```
> lserver google.com
Default server: google.com
Address: 64.233.167.99#53
Default server: google.com
Address: 64.233.187.99#53
Default server: google.com
Address: 72.14.207.99#53
```

El subcomando host ofrece una utilidad simple para realizar búsquedas. Cuando no se dan argumentos u opciones, host imprime un pequeño resumen en la línea de comando de los argumentos y las opciones. La gente lo usa principalmente para convertir nombres a direcciones IP y viceversa. He aquí un ejemplo:

```
> host centralsoft.org
centralsoft.org has address 70.253.158.42
```

Cuando pone host en modo completo con la opción -v, ofrece información similar al comando dig:

```
> host -v centralsoft.org
Trying "centralsoft.org"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43756
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;centralsoft.org.               IN     A

;; ANSWER SECTION:
centralsoft.org.       86400    IN     A      70.253.158.42

;; AUTHORITY SECTION:
centralsoft.org.       29437    IN     NS     server1.centralsoft.org.

Received 71 bytes from 68.94.156.1#53 in 30 ms
```


Esta información viene de la dirección IP 68.94.156.1, puerto 53, qué es el servidor de nombres especificado en el archivo `resolv.conf` del equipo que realizó la búsqueda. Puede usar host de nuevo para averiguar el nombre de este servidor:

```
> host 68.94.156.1
1.156.94.68.in-addr.arpa domain name pointer dnsr1.sbcglobal.net.
```

Introduzca exit para cerrar la sesión de búsqueda interactiva.

También puede usar named para arreglar fallos en algunas situaciones. Por ejemplo, para averiguar el número de versión de su implementación BIND, ejecute el siguiente comando:

```
># named -v
named 8.4.6-REL-NOESW Tue Feb 1 10:10:48 UTC 2005
      builddd@rockhopper:/build/buildd/bind-8.4.6/src/bin/named
```

rndc

BIND ofrece un comando rndc como parte de la instalación. rndc permite administrar named usando la línea de comandos. La utilidad envía los comandos introducidos mediante línea de comandos al servidor que ejecuta server, que los procesa. El script de inicialización de BIND 9 también usa rndc.

Para evitar que los usuarios no autorizados accedan a su servidor de nombres, debería usar una clave secreta para autentificar el acceso. Para que rndc ejecute los comandos en un servidor de nombres, incluso en un equipo local, ambos deben compartir la misma clave. Esta clave está almacenada en el archivo `/etc/bind/rndc.key`, y tanto named como rndc leerán la clave desde esta localización. El archivo `rndc.key` debería haberse creado durante la instalación de BIND.

El comando rndc tiene la siguiente forma:

```
#rndc rndc-options command command-options
```

Ahora veremos algunas opciones de rndc comunes que podría necesitar (lea las páginas-manual de rndc para ver la lista completa):

- **-k key-file:** Usa el archivo con la clave especificado en lugar del archivo por defecto `/etc/bind/rndc.key`.
- **-s server:** Envía el comando al servidor especificado en lugar de al servidor local.
- **-V:** Activa el modo de información completo.

Aquí se muestran algunos de los comandos que rndc suele enviar a named (para una lista completa de los comandos, simplemente introduzca el comando rndc):

- **halt:** Para el servidor de nombres inmediatamente.
- **querylog:** Activa o desactiva el log de todas las consultas hechas por los clientes a este servidor de nombres. Es un comando de conmutación: conmuta al estado activo si estaba desactivado y viceversa.
- **reload [zone]:** Recarga los archivos de zona, pero mantiene todas las respuestas que se almacenaron previamente en caché. Esto permite hacer cambios en los archivos de zona y que tengan efectos en sus servidores maestros y esclavos sin perder todos los nombres ya resueltos. Si los cambios afectan a una única zona, puede indicar que sólo se recargue esa zona.
- **retransfer zone:** Obliga a volver a transferir la zona especificada sin tener que comprobar el número de serie.
- **stats:** Vuelca las estadísticas actuales de named al archivo `named.stats`.
- **status:** Muestra el estado actual del servidor de nombres.
- **stop:** Detiene el servidor, guardando y actualizando dinámicamente los datos antes de salir.

Resolución de problemas en BIND

En este punto del capítulo, debería tener un conocimiento funcional acerca de DNS. También debería saber cómo configurar sus archivos y cómo corregir problemas de sintaxis, como errores tipográficos.

En esta sección, cubriremos algunos aspectos básicos, problemas comunes que puede encontrar cuando BIND y DNS están funcionando. No es un tratado exhaustivo, pero debería ayudarle a ejecutar DNS en su servidor Linux si tiene problemas para que su dominio resuelva nombres de equipos o haga transferencias de zona.

Nota: El diseño del sistema de nombres de dominio es robusto, pero en ocasiones pueden aparecer errores. Siguiendo estrictamente los patrones para crear archivos de zona descritos anteriormente en este capítulo, puede evitar problemas que están fuera del alcance de este libro.

No se puede conectar usando rndc

Para empezar, veamos un consejo sobre resolución DNS. Anteriormente, vimos cómo el comando status de rndc muestra el estado actual de ejecución de nuestro servidor DNS. Probemos a entrar en el sistema como root y ejecutar el comando:

```
server1:~# rndc status
number of zones: 6
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
server is up and running
server1:~#
```

El comando rndc depende de una clave compartida en el archivo `/etc/bind/rndc.key` para que named acepte sus comandos. Problemas con este archivo pueden evitar que rndc envíe los comandos.

Aquí un ejemplo de que deberíamos ver si el archivo de la clave no existiera:

```
server1:~# rndc status
rndc: neither /etc/bind/rndc.conf nor /etc/bind/rndc.key was found
server1:~#
```

Podemos comprobar que el archivo no existe con este comando:

```
server1:~# ls -l /etc/bind/rndc.key
ls: /etc/bind/rndc.key: No such file or directory
```

Podemos solucionar el problema regenerando el archivo de la misma forma que lo hace la instalación de BIND:

```
server1:~# rndc-confgen -a
server1:~# ls -l /etc/bind/rndc.key
-rw------- 1 root bind 77 Jul 19 22:38 /etc/bind/rndc.key
server1:~#
```

Debido a que named no tiene esta nueva clave, debemos matar el proceso named y reiniciarlo. Para ello, haremos uso del comando del sistema killall, que coge la ruta completa del nombre del programa named.

Para detener named de manera correcta, ejecutaremos el comando killall dos veces en un intervalo de unos cuantos segundos, luego reiniciamos named:

```
server1:~# killall -TERM /usr/sbin/named
server1:~# killall -KILL /usr/sbin/named
/usr/sbin/named: no process killed
server1:~# /etc/init.d/bind9 start
Starting domain name service: named.
server1:~# rndc status
number of zones: 6
```

```
debug level: 0
xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
server is up and running
server1:~#
```

named se inicia pero no resuelve nombres

Ahora, veamos algunas situaciones donde named no funciona correctamente. La localización incorrecta de los archivos BIND a menudo causa problemas, especialmente en entornos chroot donde los archivos BIND están en un directorio aislado. Si named se inicia bien pero no carga ningún archivo de zona, puede que no estén en el directorio aislado. Necesitará mirar el archivo `/var/log/syslog` para ver si este es el caso. He aquí un ejemplo de log:

```
starting BIND 9.2.4 -u bind -t /var/lib/named
using 1 CPU
loading configuration from '/etc/bind/named.conf'
listening on IPv4 interface lo, 127.0.0.1#53
listening on IPv4 interface eth0, 70.253.158.42#53
command channel listening on 127.0.0.1#953
command channel listening on ::1#953
running
```

El log muestra que BIND se ha iniciado, pero no incluye líneas indicando que los archivos de zona se han cargado. Ya que named se ejecuta en un entorno chroot en `/var/lib/named`, buscará todos los archivos relativos del directorio. Por lo que realmente está leyendo el archivo `/var/lib/named/etc/bind/named.conf` para la lista de zonas que debe cargar. Cada uno de estos archivos de zona debe colocarse en la ruta relativa del directorio `/var/lib/named`.

Otro error común es el fallo de una conexión que involucra a rndc al recargar o reiniciar el servidor de nombres:

```
#/etc/init.d/ bind9 reload
Stopping named: rndc: connect failed: connection refused
[OK]
Starting named: [OK]
#
```

Este tipo de error también puede suceder como resultado de ejecutar BIND en un entorno chroot, cuando uno o más archivos no está en el directorio aislado. Puede comprobar si los archivos están en las localizaciones correctas así:

```
# ls -l /var/lib/named/etc/bind/named.conf
-rw-r--r-- 1 root bind 1611 2006-09-07 12:21 /var/lib/named/etc/bind/
named.conf
```



```
# ls /var/lib/named/etc/bind/
db.0      db.local    named.conf.local                pri.centralsoft.org
db.127    db.root     named.conf.options              pri.opensourcetoday.
                                                      org
db.255    named.conf  pri.156.18.67.in-addr.arpa      rndc.key
db.empty  named.conf~ pri.156.18.67.in-addr.arpa~     zones.rfc1918
#
...
```

Si estos archivos no existen, el entorno chroot no está configurado adecuadamente o completamente. Vuelva al comienzo del capítulo y siga las instrucciones con cuidado para asegurarse de que cada archivo está en su lugar.

Una vez arreglado el problema, necesitará parar y reiniciar named para que rndc pueda conectar con el servidor. Use la secuencia de comandos killall descrita en la sección previa:

```
server1:~# killall -TERM /usr/sbin/named
server1:~# killall -KILL /usr/sbin/named
/usr/sbin/named: no process killed
server1:~# /etc/init.d/bind9 start
Starting domain name service: named.
server1:~#
```

Ahora, compruebe el archivo `/var/log/syslog` para ver si los archivos de zona se han cargado. Debería ver algo como esto:

```
starting BIND 9.2.4 -u bind -t /var/lib/named
using 1 CPU
loading configuration from '/etc/bind/named.conf'
listening on IPv4 interface lo, 127.0.0.1#53
listening on IPv4 interface eth0, 70.253.158.42#53
command channel listening on 127.0.0.1#953
command channel listening on ::1#953
zone 0.0.127.in-addr.arpa/IN: loaded serial 1
zone 158.253.70.in-addr.arpa/IN: loaded serial 2006070401
zone centralsoft.org/IN: loaded serial 2006070502
zone supportcall.org/IN: loaded serial 2006062704
running
```

No se reconocen los equipos

El siguiente paso es comprobar el correcto funcionamiento de DNS para asegurarse de que las consultas acerca de sus equipos se responden de manera adecuada. Primero, necesita asegurarse de que el archivo `/etc/resolv.conf` lista sus servidores de nombres con las direcciones correctas. La mayoría de los programas usan las direcciones de este archivo para determinar qué servidores de nombres deben consultar y en qué orden:

```
server1:~# cat /etc/resolv.conf
search centralsoft.org
nameserver 70.253.158.42
nameserver 70.253.158.45
server1:~#
```

El comando host hace una simple búsqueda DNS usando los servidores listados en el archivo `/etc/resolv.conf`. Necesita el equipo a buscar como parámetro, y un segundo parámetro opcional hace que el comando consulte a un servidor de nombres específico. He aquí dos ejemplos del comando host y sus resultados:

```
server1:~# host www.centralsoft.org
www.centralsoft.org has address 70.253.158.42
server1:~# host www.centralsoft.org server2.centralsoft.org
Using domain server:
Name: server1.centralsoft.org
Address: 70.253.158.45#53
Aliases:

www.centralsoft.org has address 70.253.158.42
server1:~#
```

Una alternativa a host es el comando dig, que es más complejo pero ofrece respuestas más detalladas. También tiene más opciones que le posibilitan realizar consultas más específicas. La salida de dig está formateada según la sintaxis del archivo de zona. Esto es una ventaja, puesto que una vez que haya aprendido qué formato tienen los registros en un archivo de zona, puede comprender fácilmente los detalles de estos registros en la salida de dig. dig también ofrece información adicional sobre los resultados de la consulta en los comentarios que empiezan por el carácter ";".

Echemos un vistazo al resultado del comando dig. Muchas líneas de la salida de dig son muy largas y no caben en el diseño de página de este libro. En el siguiente listado, las hemos divido en líneas. Podrá ver un resultado similar cuando lo ejecute en su línea de comandos:

```
server1:~# dig www.centralsoft.org a

; <<>> DiG 9.2.4 <<>> www.centralsoft.org a
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1633
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;;www.centralsoft.org.            IN    A

;; ANSWER SECTION:
```



```
www.centralsoft.org.       86400    IN    A     70.253.158.42

;; AUTHORITY SECTION:
centralsoft.org.           86400    IN    NS    server1.centralsoft.org.
centralsoft.org.           86400    IN    NS    server2.centralsoft.org.

;; ADDITIONAL SECTION:
server1.centralsoft.org.   86400    IN    A     70.253.158.42
server2.centralsoft.org.   86400    IN    A     70.253.158.45

;; Query time: 1 msec
;; SERVER: 70.253.158.42#53(70.253.158.42)
;; WHEN: Mon Jul 17 23:30:51 2006
;; MSG SIZE rcvd: 129
server1:~#
```

La primera parte de la salida indica varios códigos de estado y banderas. Preste atención particular al valor del estado de la cuarta línea. Es este ejemplo, el valor es NOERROR. Cualquier otro valor indica algún tipo de problema.

Los datos para la zona actual se dividen en cuatro secciones:

- QUESTION: Esta sección actualmente detalla una consulta. Se muestra como un comentario porque no es la información que debería estar en un archivo de zona.
- ANSWER: Esta sección contiene los resultados actuales solicitados por la consulta. Mostrará los registros específicos solicitados, si están disponibles, o todos los registros si se está usando el tipo de consulta especial.
- AUTHORITY: Esta sección identifica los servidores de nombres oficiales para la zona de la que viene la transferencia.
- ADDITIONAL: Esta sección ofrece las direcciones de algunos o de todos los nombres de las secciones anteriores, para evitar el problema de hacer más consultas para obtener esa información. Ahora, veamos lo que debería hacer si se produjera un error. El ejemplo anterior usaba un nombre de equipo válido para el servidor Web. Esta vez lanzaremos una consulta para el nombre de un servidor FTP que no hayamos configurado en nuestro archivo de zona:

```
server1:~# dig ftp.centralsoft.org a

; <<>> DiG 9.2.4 <<>> ftp.centralsoft.org a
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 6531
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;ftp.centralsoft.org.        IN    A
```

```
;; AUTHORITY SECTION:
centralsoft.org.     86400    IN    SOA   server1.centralsoft.org. admin.
centralsoft.org. 2006070502  28800 7200  604800    86400

;; Query time: 1 msec
;; SERVER: 70.253.158.42#53(70.253.158.42)
;; WHEN: Mon Jul 17 23:30:59 2006
;; MSG SIZE rcvd: 87

server1:~#
```

Fíjese en que el estado para la consulta es NXDOMAIN, que en esencia significa "no existe tal nombre de dominio". Si usted no tiene o introduce mal el nombre de equipo en el archivo de zona, obtendrá este error.

Otro tipo de error que podría ver con dig es cuando un nombre de dominio ha sido delegado a su servidor de nombres, pero el dominio no está configurado en el servidor o falla al cargarse. Este error devuelve un estado de SERVFAIL. Si ve este error para alguno de sus dominios, necesita añadir el dominio a su archivo named.conf y asegurar que existe un archivo de zona válido. Si el error ocurre después de haber dado estos pasos, compruebe el archivo /var/log/syslog para ver los mensajes que indican por qué no se ha cargado la zona. Demostraremos el problema con un nombre de dominio que está registrado, pero que no está en uso:

```
server1:~# dig linhelp.org a

; <<>> DiG 9.2.4 <<>> linhelp.org a
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 29949
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;linhelp.org.        IN    A

;; Query time: 2 msec
;; SERVER: 70.253.158.42#53(70.253.158.42)
;; WHEN: Mon Jul 17 23:47:14 2006
;; MSG SIZE rcvd: 37

server1:~#
```

Qué es lo próximo

Por ahora, ya debería estar familiarizado con los aspectos básicos de DNS y BIND. Los administradores de negocios de pequeño y mediano tamaño deberían

encontrar en este capítulo todo lo que necesitan, pero los administradores de sistemas empresariales seguro que se encontrarán con aspectos más complejos de los que puede tratar un único capítulo de un libro.

Hay algunos libros que pueden ofrecer información mucho más detallada para administradores DNS de grandes empresas. Entre estos libros destacan "DNS y BIND" de Cricket Liu y Paul Albitz (O'Reilly), "DNS & BIND Cookbook" de Cricket Liu (O'Reilly), "Pro DNS and BIND" de Ron Aitchison (Apress) y "DNS in Action: A Detailed and Practical Guide to DNS Implementation, Configuration, and Administration" de L. Dostalek y A. Kableova (Packt).

Ahora que ya tiene un servidor de nombres respondiendo consultas y un servidor secundario o esclavo que actúa como copia de seguridad, en el próximo capítulo podrá aprender a instalar una aplicación de servicios Web. Esta nueva aplicación usará los servicios configurados en el capítulo anterior. Una vez que la aplicación llamada ISPConfig esté configurada y se esté ejecutando, podrá ver un ejemplo de un sitio Web completamente operativo. Podemos empezar explorando cómo administrar la suite completa de los servicios para Internet que ofrece Linux.

Capítulo 4

Un entorno inicial listo para Internet



Una de las grandes ventajas de Linux es su flexibilidad. Las compañías comerciales como Cisco han ocultado Linux debajo de interfaces simples que hacen de sus routers Linksys productos amigables. Nosotros también podemos hacer esto.

ISPConfig (`http://ispconfig.org`), un Linux amigable bajo licencia de software libre (BSD), permite levantar servidores multifuncionales que trabajen en Internet desde una aplicación disponible para descargar. Una vez que lo hemos instalado, dispondremos de una herramienta que nos ayuda a configurar y mantener fácilmente un servidor que nos permite gestionar sitios Web; ofrecer servicios de nombres de dominio, realizar transferencias de archivos y de correo, y añadir usuarios, administradores y otros para que puedan acceder al sistema y realizar tareas administrativas. Además, ¿hemos mencionado que permite realizar todas estas tareas de administración desde una interfaz gráfica?

Hemos seleccionado ISPConfig principalmente porque nos permite desplegar servidores de aplicaciones muy potentes sobre Linux sin sacrificar su potencia o su flexibilidad. Además:

- ISPConfig usa demonios estándar que vienen con las distribuciones Linux. Usaremos Apache para servir sitios Web. Postfix para el correo electrónico. ProFTPD para el FTP, BIND para el DNS y MySQL como gestor de Bases de datos.
- La instalación de ISPConfig configura de manera automática varios servidores de componentes.
- Los paquetes incluidos en ISPConfig funcionan con la mayoría de las distribuciones Linux disponibles.
- Pueden usarse paquetes estándar de las distribuciones.

- Puede encontrarse soporte en Internet para cada componente.
- El equipo de ISPConfig ofrece soporte en línea para la aplicación entera.

A medida que avance en el capítulo, debería ir haciéndose una idea de lo que un servidor necesita para ofrecer varios servicios y funcionar. También aprenderá cómo decidir si necesita usar un panel administrativo visual en lugar de una interfaz de línea de comandos.

ISPConfig no ofrece una interfaz de línea de comandos. En su lugar, le permite gestionar a través de una interfaz administrativa basada en Web, o un panel, que se describirá más adelante en este capítulo. Tendrá que ejecutar algunas órdenes desde la línea de comandos al principio del capítulo, al configurar ISPConfig puesto que no instala todo secuencialmente, pero en las siguientes secciones, nos centraremos exclusivamente en la interfaz visual.

El panel Web de ISPConfig simplifica la ejecución de muchas tareas administrativas de Linux, pero es importante saber cómo usar las utilidades de línea de comando para obtener los mismos resultados. Cubriremos estos aspectos en los capítulos posteriores. Esto no supone que tenga que estar atado a ISPConfig, pero si elige trabajar sin él, es conveniente que sepa para qué sirve.

Instalando ISPConfig

ISPConfig viene del Projektfarm GmbH. La aplicación fue desarrollada por Till Brehm y Falko Timme, que originalmente la vendieron como una aplicación propietaria en `http://42go.de`. Ahora puede descargarla desde `http://sourceforge.net/projects/ispconfig`.

El proyecto configura estos servicios:

- httpd (equipos virtuales, basados en dominio y basados enIP).
- FTP.
- BIND.
- POP3 de autorespuesta.
- Cliente de MySQL.
- Estadísticas Webalizer.
- Cuotas de disco duro.
- Cuotas de correo.
- Límites de tráfico.
- Direcciones IP.
- SSL.

- SSI.
- Acceso a la línea de comandos.
- Escáner de correo (antivirus).
- Firewall.

Requisitos

En este momento, los requisitos del sistema son:

- Sistema Operativo: Linux (kernel 2.4 o posterior con la librería glibc6).

  Las siguientes distribuciones lo permiten:
  - CentOS 4.1, 4.2, 4.3 y 4.4.
  - Debian Versión 3.0 o posterior.
  - Fedora Core de la 1 a la 6.
  - Mandrake Linux Versión 8.1 o posterior.
  - Mandriva 2006 y 2007.
  - Red Hat Linux Versión 7.3 o posterior.
  - SUSE Linux Version 7.2 o posterior.
  - Ubuntu desde 5.04 a la 6.10.
- Paquetes Linux: Los mantenedores del proyecto aseguran que son necesarios varios componentes en el sistema antes de instalar ISPConfig. Estos componentes son:
  - Apache web server Version 1.3.12 o posterior, o 2.0.40 o posterior.
  - BIND 8 o 9.
  - iptables o ipchains.
  - MySQL.
  - OpenSSL y mod_ssl para la creación de host virtuales.
  - PHP 4.0.5 o posterior como modulo Apache.
  - POP3/IMAP, demonio que soporta el formato de buzón tradicional de Unix (por ejemplo, gnu-pop3d, qpopper, ipop3d, popa3d o vm-pop3d) o el format maildir (por ejemplo, Courier-Imap, Dovecot).
  - Procmail.
  - ProFTP como versión independiente o vsftpd como versión inetd/xinetd/independiente.
  - Paquete de cuotas.
  - Sendmail o Postfix.

Es importante comprender que estos servidores y paquetes deben estar instalados en el sistema como se describió anteriormente, antes de instalar ISPConfig. ISPConfig no viene con estos servicios, pero necesita que existan en el sistema. La ventaja de esta solución es que puede usar los paquetes por defecto de su distribución y actualizarlos después usando las herramientas de su distribución. No tiene que compilar estos servicios a partir de las fuentes con opciones específicas para ISPConfig, los paquetes por defecto ya lo hacen.

ISPConfig configura dos directorios que contienen los archivos y los subdirectorios que integran el panel de aplicación: `/root/ispconfig` y `/home/admispconfig`. Puede desinstalar ISPConfig y volver al servidor basado en texto ejecutando `/root/ispconfig/uninstall`; algunos lectores harán esto después de leer este libro.

Demonios especiales de ISPConfig

Además de gestionar las aplicaciones que ha instalado en el sistema, ISPConfig mantiene sus propias versiones de unas cuantas aplicaciones que necesita usar. Puede encontrar las fuentes de estas aplicaciones en el directorio `install_ispconfig/compile_aps`. Estos servicios redundantes existen, por lo que puede continuar gestionando ISPConfig incluso si los servicios normales (como el servidor público Apache) se caen.

ISPConfig permite tanto a los servidores públicos como a los internos funcionar usando un puerto no estándar. Por ejemplo, el servidor interno de Apache para ISPConfig escucha en el puerto 81 en lugar de en el 80, que es el utilizado normalmente por el servidor Web para ofrecer sitios Web públicos.

Comenzando

Al igual que muchos paquetes Linux y Unix, ISPConfig se ofrece como un conjunto de archivos comprimidos con la utilidad tar, el resultado de ello a menudo se llama tarball. Al hacer clic en el enlace Download de `http://sourceforge.net/projects/ispconfig`, le dirige a uno de los mirrors de SourceForge. Un sitio que normalmente tiene ISPConfig es `http://superb-west.dl.sourceforge.net/sourceforge/ispconfig/ISPConfig-2.2.6.tar.gz`.

Puede hacer clic en el enlace Download para descargar el archivo, pero debido a que el archivo es muy grande, le será más útil copiar la URL y pegarla en una ventana de terminal a la hora de invocar el comando wget. La ventaja de usar wget es que un archivo se puede recuperar a pesar de que se haya interrumpido la descarga. Si usa el comando con la opción –c, puede reanudar la descarga sin tener que volver a empezar: si la descarga se interrumpe, simplemente vuelva a ejecutar la orden wget y la descarga se retomará por donde se dejó.

En este capítulo supondremos que comienza en un directorio de su sistema llamado `/root`. Puede descargar la distribución ISPConfig con este comando (en una línea, sustituyendo la URL por la versión más reciente ofrecida en el sitio SourceForge):

```
# wget -c http://superb-west.dl.sourceforge.net/sourceforge/ispconfig/
ISPConfig-2.2.
6.tar.gz
```

Su terminal mostrará mensajes parecidos al siguiente:

```
--16:20:48--  http://superb-west.dl.sourceforge.net/sourceforge/
ispconfig/ISPConfig-2.2.1.tar.gz
        => 'ISPConfig-2.2.1.tar.gz'
Resolving superb-west.dl.sourceforge.net... 209.160.59.253
Connecting to superb-west.dl.sourceforge.net|209.160.59.253|:80...
connected.
HTTP request sent, awaiting response... 200 OK
 Length: 26,633,490 (25M) [application/x-gzip]
24% [========>          ] 6,533,049    252.80K/s      ETA 01:32
```

Descomprima el archivo con el comando:

```
# tar xvfz ISPConfig*.tar.gz
```

que crea un directorio llamado `install_ispconfig`. Muévase del directorio `/root/ al install_ispconfig`. Compruebe el archivo `dist.txt` y vea si los valores son apropiados para su servidor Linux.

Para Debian 3.1, los valores de dist.txt serían:

```
dist_init_scripts=/etc/init.d     ## # debian31
dist_runlevel=/etc                ## # debian31
```

El archivo contiene 19 valores adicionales para Debian que no están listados aquí. A menos que tenga cierta experiencia en administración Linux y esté familiarizado con ISPConfig, respete los valores por defecto. Debería funcionar siempre y cuando esté usando una de las distribuciones que se listaban con anterioridad. Los administradores con conocimientos pueden cambiar los valores, aunque respetando el formato del archivo.

Ahora inicie la instalación. Ejecute el comando de instalación `./setup` desde la línea de comandos. El script de instalación empezará a compilar Apache con PHP 5 que escuchará en el puerto 81. Primero, tendrá que decidir qué idioma quiere:

```
server2:~/install_ispconfig # ./setup
SuSE 10.0
Neu installation eines ISPConfig-Systems. / Installation of a new
ISPConfig system. /
```

```
Installation d'ISPConfig sur un nouveau systeme.
Whlen Sie Ihre Sprache (deutsch/englisch/spanisch/franzsisch/italienisch/
niederlndisch/polnisch/
schwedisch): / Please choose your language (German/English/Spanish/
French/Italian/
Dutch/Polish/Swedish): / Merci de choisir votre langue (Allemand/
Anglais/Espagnol/
Francais/Italien/Nerlandais/Polonais/Sudois):
1) de
2) en
3) es
4) fr
5) it
6) nl
7) pl
8) se
Ihre Wahl: / Your Choice: / Votre Choix:
```

Verá una pantalla de advertencia:

```
With the system installation, some system files are replaced where
adjustments were
made. This can lead to loss of entries in httpd.conf, named.conf as
well as in the
Sendmail configuration.
Do you want to continue with the installation? [y/n] y
```

El sistema mostrará una licencia, debería leerla y aceptarla:

```
Do you accept the license? [y/n] y
```

El programa de instalación procederá a preguntarle cuestiones acerca de la configuración del sistema (p. Ej, por el MTA, servidor FTP, servidor Web, logs, etc...) debido a que tiene que tener instalados todos los paquetes en su sistema, debería ser capaz de responder a todas las preguntas.

Durante la primera parte de la instalación, el script le preguntará en qué modo quiere ejecutar la instalación. Seleccione el modo experto:

```
1) standard
2) expert
Your Choice: 2
```

En modo experto, tendrá que elegir algunas opciones que ISPConfig asigna por defecto en el modo estándar.

```
   When prompted for a default directory, you can choose any directory
you like, but make sure it is on a partition with enough disk space for
the web sites you plan to host. Furthermore, if you want to configure
quotas with ISPConfig, make sure you enabled quotas for that partition
as described in Chapter 2. If you want to enable suExec for web sites
```



```
that are allowed to run Perl/CGI scripts, the directory should be within
suExec's document root. On Debian and Fedora/Red Hat, suExec's default
document root is /var/www, while on SUSE it's /srv/www. If you're
enabling suExec, the document root is a good choice for the directory in
which to put ISPConfig:
   ########## WEB SERVER ##########
   Checking for program httpd...
   /usr/sbin/httpd
   OK
   Checking the syntax of the httpd.conf...
   Syntax OK
   The syntax is ok!
   Web-Root: /home/www
   Is this correct? [y/n] n
   Web-Root: /var/www
```

Nota: suExec es una mejora de seguridad en un servidor Web que necesita scripts CGI para poder ejecutarse por parte de ciertos usuarios.

En este punto la instalación comienza compilando el servidor Apache que se usará para presentar la interfaz Web ISPConfig en el puerto 81. Cuando el Apache para ISPConfig esté completo, verá un certificado SSL compilado. El programa de instalación le preguntará varios valores. Puede aceptar los valores por defecto o introducir los suyos propios. La pantalla será similar a esta:

```
SSL Certificate Generation Utility (mkcert.sh)
Copyright (c) 1998-2000 Ralf S. Engelschall, All Rights Reserved.
Generating custom certificate signed by own CA [CUSTOM]
The system will display a license, which you should read and then accept:
Do you accept the license? [y/n] y
______________________________________________________________________
STEP 0: Decide the signature algorithm used for certificates
The generated X.509 certificates can contain either
RSA or DSA based ingredients. Select the one you want to use.
Signature Algorithm ((R)SA or (D)SA) [R]:
______________________________________________________________________
STEP 1: Generating RSA private key for CA (1024 bit) [ca.key]
1698765 semi-random bytes loaded
Generating RSA private key, 1024 bit long modulus
...........++++++
.........++++++
e is 65537 (0x10001)
______________________________________________________________________
STEP 2: Generating X.509 certificate signing request for CA [ca.csr]
You are about to be asked to enter information that will be incorporated
 into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
```

```
If you enter '.', the field will be left blank.
1. Country Name               (2 letter code)    [XY]:
2. State or Province Name     (full name)        [Snake Desert]:
3. Locality Name              (e.g, city)        [Snake Town]:
4. Organization Name          (e.g, company)     [Snake Oil, Ltd]:
5. Organizational Unit Name   (e.g, section)     [Certificate Authority]:
6. Common Name                (eg, CA name)      [Snake Oil CA]:
7. Email Address              (e.g, name@FQDN)   [ca@snakeoil.dom]:
8. Certificate Validity       (days)             [365]:
______________________________________________________________________
STEP 3: Generating X.509 certificate for CA signed by itself [ca.crt]
Certificate Version (1 or 3) [3]:
Signature ok
subject=/C=XY/ST=Snake Desert/L=Snake Town/O=Snake Oil, Ltd/OU=Certificate
Authority/
CN=Snake Oil CA/emailAddress=ca@snakeoil.dom
Getting Private key
Verify: matching certificate & key modulus
Verify: matching certificate signature
../conf/ssl.crt/ca.crt: /C=XY/ST=Snake Desert/L=Snake Town/O=Snake Oil, Ltd/
OU=Certificate Authority/CN=Snake Oil CA/emailAddress=ca@snakeoil.dom
error 18 at 0 depth lookup:self signed certificate
OK
______________________________________________________________________
STEP 4: Generating RSA private key for SERVER (1024 bit) [server.key]
1698765 semi-random bytes loaded
Generating RSA private key, 1024 bit long modulus
..........................++++++
............++++++
e is 65537 (0x10001)
______________________________________________________________________
STEP 5: Generating X.509 certificate signing request for SERVER [server.csr]
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
1. Country Name               (2 letter code)    [XY]:
2. State or Province Name     (full name)        [Snake Desert]:
3. Locality Name              (eg, city)         [Snake Town]:
4. Organization Name          (eg, company)      [Snake Oil, Ltd]:
5. Organizational Unit Name   (eg, section)      [Webserver Team]:
6. Common Name                (eg, FQDN)         [www.snakeoil.dom]:
7. Email Address              (eg, name@fqdn)    [www@snakeoil.dom]:
8. Certificate Validity       (days)             [365]:
______________________________________________________________________
STEP 6: Generating X.509 certificate signed by own CA [server.crt]
Certificate Version (1 or 3) [3]:
Signature ok
subject=/C=XY/ST=Snake Desert/L=Snake Town/O=Snake Oil,Ltd/
OU=WebserverTeam/CN=www.
```



```
snakeoil.dom/emailAddress=www@snakeoil.dom
Getting CA Private Key
Verify: matching certificate signature
../conf/ssl.crt/server.crt: OK
```

En los pasos 7 y 8 del proceso de creación del certificado, se le preguntará si quiere encriptar las claves respectivas:

```
______________________________________________________________________
STEP 7: Enrypting RSA private key of CA with a pass phrase for security
[ca.key]
The contents of the ca.key file (the generated private key) has to be
kept secret. So we strongly recommend you to encrypt the server.key file
with a Triple-DES cipher and a Pass Phrase.
Encrypt the private key now? [Y/n]: n
writing RSA key
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
Fine, you're using an encrypted private key.
______________________________________________________________________
STEP 8: Enrypting RSA private key of SERVER with a pass phrase for
security [server.key]
The contents of the server.key file (the generated private key) has to be
kept secret. So we strongly recommend you to encrypt the server.key file
with a Triple-DES cipher and a Pass Phrase.
Encrypt the private key now? (Y/n): n
What email address or URL should be used in the suspected-spam report
text for users
who want more information on your filter installation?
(In particular, ISPs should change this to a local Postmaster contact)
default text: [the administrator of that system]
```

Responda *n* a estas preguntas. En caso contrario, siempre se le preguntará la contraseña cuando quiera reiniciar el sistema ISPConfig, lo que significa que no podrá reiniciarse sin interacción humana.

Si la compilación falla, el proceso de instalación se para y todos los archivos compilados se eliminan. El mensaje de error que obtenga debería indicar la razón del fallo. En la mayoría de los casos, los archivos de cabecera para un paquete son los que faltan.

Sabiendo la razón, revise la configuración del servidor y solucione el problema. *Si el directorio* `install_ispconfig` *no se ha eliminado, bórrelo manualmente.* Luego, descomprima las fuentes de nuevo, vaya al nuevo directorio `install_ispconfig` y ejecute `./setup`. No puede instalar ISPConfig dos veces desde el mismo directorio `install_ispconfig` después de que haya ocurrido algún error.

Si alguno de los paquetes necesarios no está presente, la rutina de instalación se detendrá. Instale el paquete perdido, borre el directorio `install_ispconfig`, descomprima `ISP_Config` de nuevo y vuelva a comenzar.

El script de instalación verifica la sintaxis de los archivos de configuración de Apache existentes. Un error provocará que la instalación de ISPConfig se detenga.

Si todas las condiciones se cumplen, necesitará ofrecer los siguientes valores durante la instalación:

```
Please enter your MySQL server: localhost
Please enter your MySQL user: root
Please enter your MySQL password: Su contraseña para MySQL
Please enter a name for the ISPConfig database: ispconfigdb
Please enter the IP address of the ISPConfig web: 192.168.0.1
Please enter the host name: www
Please enter the domain: xyz.de
```

Luego el programa de configuración le preguntará qué protocolo quiere usar. Seleccione el 2, HTTP:

```
Please select the protocol (http or https (SSL encryption)) to use to
access the
ISPConfig system:
1) HTTPS
2) HTTP
Your Choice: 2
```

Verá que el sistema ejecuta los script finales y reinicia algunos servicios:

```
Connected successfully to MySQL server
ls: /etc/apache2/vhosts.d/*.conf: No such file or directory
Restarting some services...
which: no apachectl in (/sbin:/usr/sbin:/usr/local/sbin:/root/bin:/
usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/usr/local/libexec)
Shutting down mail service (Postfix)                done
Starting mail service (Postfix)                     done
Shutting down mail service (Postfix)                done
Starting mail service (Postfix)                     done
Shutting down ProFTPD Server:                       done
Starting ProFTPD Server: - warning: "ProFTPD" address/port (70.253.158.45:21)
already in use by "ProFTPD Default Installation"
                                                    done
Shutting down ProFTPD Server:                       done
Starting ProFTPD Server: - warning: "ProFTPD" address/port (70.253.158.45:21)
already in use by "ProFTPD Default Installation"
                                                    done
Starting ISPConfig system...
/root/ispconfig/httpd/bin/apachectl startssl: httpd started
ISPConfig system is now up and running!
```

Los desarrolladores finalizan el script de instalación con:

```
Congratulations! Your ISPConfig system is now installed. If you had to
install quota, please take the steps described in the installation manual.
Otherwise your system is now available without reboot.
```


Llegados a este punto, puede indicar la dirección IP de su servidor y el nombre de dominio seguido por :81 en su navegador para acceder a la pantalla de login de ISPConfig.

Estructura de directorios de ISPConfig

Como se mencionó previamente, el directorio principal configurado por ISPConfig se llama ispconfig y está situado en el directorio donde se instaló (/root en este capítulo). También puede encontrar otro directorio en /home llamado admisp-config. Cada directorio contiene los archivos necesarios para ejecutar ISPConfig independientemente.

Echemos un vistazo al directorio /root/ispconfig:

```
-rwxr-xr-x   1 root  root  33660 2006-04-26  12:28 cronolog
-rwxr-xr-x   1 root  root   9673 2006-04-26  12:28 cronosplit
drwxr-xr-x  12 root  root   4096 2006-04-26  09:55 httpd
drwxr-xr-x  12 root  root   4096 2006-04-26  12:28 isp
-rw-r--r--   1 root  root      8 2006-04-26  13:54 .old_path_httpd_root
drwxr-xr-x   6 root  root   4096 2006-04-26  09:50 openssl
drwxr-xr-x   6 root  root   4096 2006-04-26  10:00 php
drwxr-xr-x   4 root  root   4096 2006-04-26  12:28 scripts
drwxr-xr-x   4 root  root   4096 2006-04-26  12:28 standard_cgis
drwxr-xr-x   2 root  root   4096 2006-04-26  12:28 sv
-rwx------   1 root  root   9389 2006-04-26  12:28 uninstall
```

Contiene los diferenntes archivos de configuración de ISPConfig, Apache, PHP y OpenSSL, así como varias plantillas para todos los tipos de archivos de configuración (para Apache, Postfix, Sendmail, BIND, procmail, etc.) ISPConfig usa las plantillas para poder escribir los archivos de configuración para los servicios que configura.

También encontrará muchas clases PHP que ofrecen funciones para escribir en los archivos de configuración del sistema. Resumiendo, /root/ispconfig contiene el núcleo de ISPConfig. En el directorio /home/admispconfig, podrá ver otro conjunto de directorios:

```
-rwxr-xr-x  1 admispconfig admispconfig   24 2006-04-26 12:28 .forward
drwxr-xr-x  8 admispconfig admispconfig 4096 2006-04-26 13:53 ispconfig
drwxr-xr-x  2 admispconfig admispconfig 4096 2006-04-26 12:28 mailstats
-rwxr-xr-x  1 admispconfig admispconfig  176 2006-04-26 12:28 .procmailrc
```

Contiene la interfaz de ISPConfig, es decir la interfaz Web, además de algunas herramientas como SpamAssassin (http://spamassassin.apache.org) y ClamAV (http://clamav.elektrapro.com). Puede configurar estos a través de ISPConfig para protegerse contra el spam y los virus.

Configurando un servidor y usuarios con ISPConfig

Configurar un sitio Web es uno de los primeros pasos para conseguir un servidor de Internet completamente funcional. Esta sección le mostrará todos los pasos necesarios.

Tal vez piense que puede ir al sitio Web de ISPConfig y leer los manuales, sin embargo tenga en cuenta que los desarrolladores de ISPConfig escribieron la documentación de usuario para sitios dedicados al alojamiento Web. Si esta es su intención, le recomendamos que lea los manuales de `http://ispconfig.org`. En caso contrario, supondremos que desea usar su servidor con un único administrador del sistema que gestiona sus propios sitios Web seguros, el correo y los servicios FTP.

ISPConfig necesita que configure un cliente que tenga uno o más dominios de Internet. En nuestro ejemplo, configuraremos un cliente sencillo (de uno de los autores del libro) que tiene cuatro dominios:

- `centralsoft.org`
- `linuxnewswire.org`
- `opensourcetoday.org`
- `tadelstein.com`

Cuando mire el contenido del directorio `/var/www` verá cómo ISPConfig configura los siguientes dominios:

```
$ ls -a
apache2-default sharedip web2 web4 webalizer www.opensourcetoday.org
localhost        web1     web3 www.centralsoft.org www.linuxnewswire.org
www.tadelstein.com
```

Compare el listado de este directorio con la lista de sitios Web de la figura 4.1. Cada sitio Web contiene un directorio. El directorio www cuyos nombres muestran los dominios (como `www.opensourcetoday.org`) son enlaces simbólicos que el sistema conoce como web1, web2, etc.

La figura 4.2 le da una mejor idea de la lista de dominios. Fíjese que en la figura 4.2 el domino aparece para cada directorio del listado de la línea de comandos.

Añadiendo clientes y sitios Web

Para configurar el cliente y los dominios, primero debe entrar en la interfaz de ISPConfig. En su navegador Web, teclee la dirección IP del servidor seguida del puerto para ISPConfig :81. En nuestro caso, `http://70.253.158.45:81` (use

https:// si seleccionó HTTPS como el protocolo ISPConfig durante la instalación). En la pantalla de autentificación (figura 4.3), introduzca el ID admin y la contraseña admin.

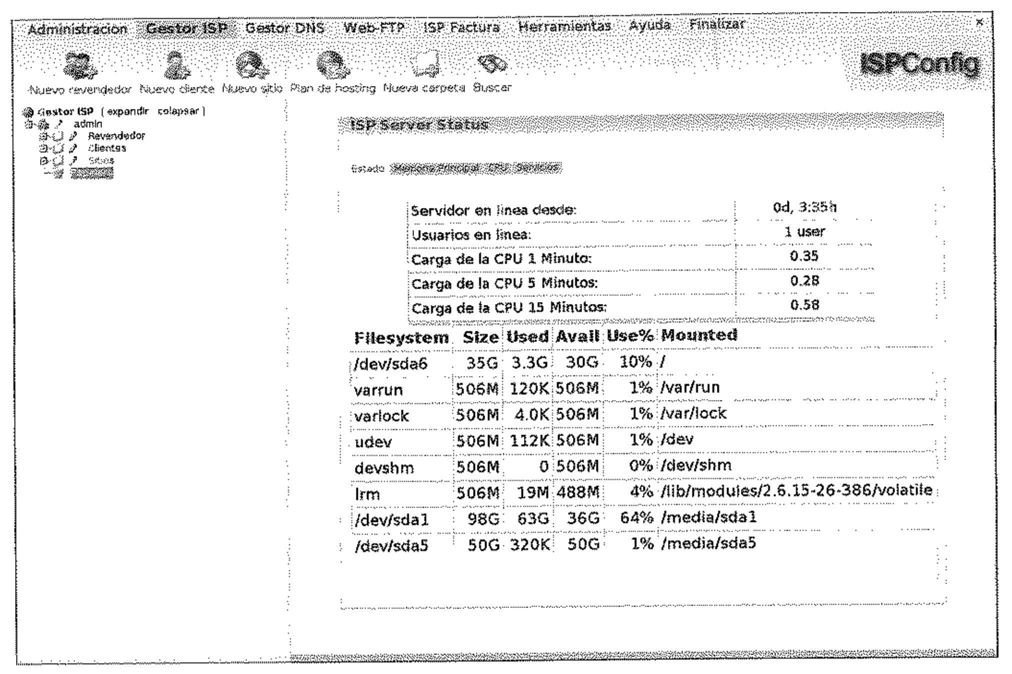


Figura 4.1. Interfaz de gestión de ISP.

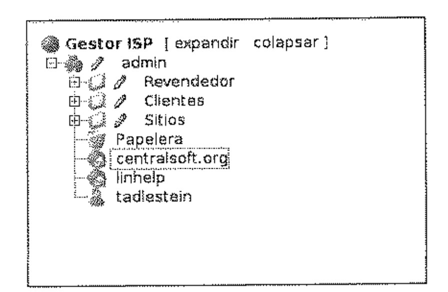


Figura 4.2. Lista de dominios que gestiona ISP.

Luego, inmediatamente cambie la contraseña por una que sólo usted conozca. Para poder cambiar la contraseña seleccione Herramientas desde la barra de herramientas y seguidamente haga clic en el símbolo de contraseña (véase la figura 4.4).

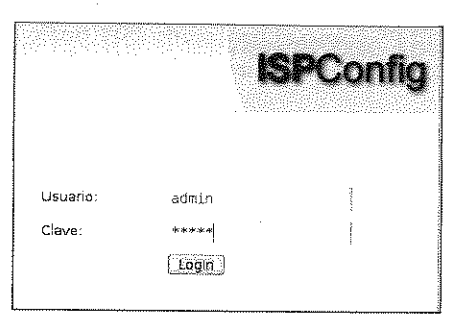


Figura 4.3. Pantalla de autentificación de ISPConfig.

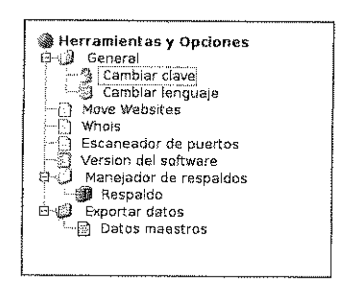


Figura 4.4. Menú Herramientas.

En la figura 4.5 se muestra el cuadro de diálogo Cambiar contraseña, allí podrá rellenar el formulario.

Salga del sistema y entre con la nueva contraseña.

Figura 4.5. Formulario de ISPConfig para cambiar contraseñas.

Antes de configurar un sitio Web, tendrá que crear un propietario del sitio. Seleccione la barra de herramientas Gestión de ISP. Verá un menú de navegación similar al de la figura 4.6.

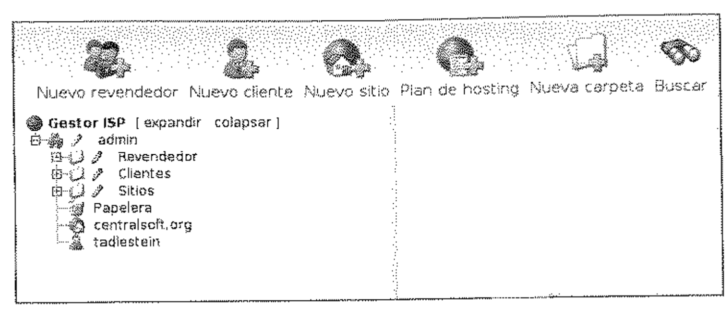


Figura 4.6. El menú de gestión de ISP con un cliente y un dominio.

Ahora veamos cómo hemos creado el cliente tadelstein y el sitio Web linhelp. Haga clic en Nuevo Cliente en el menú de gestión de ISP. Verá un cuadro de diálogo similar al de la figura 4.7.

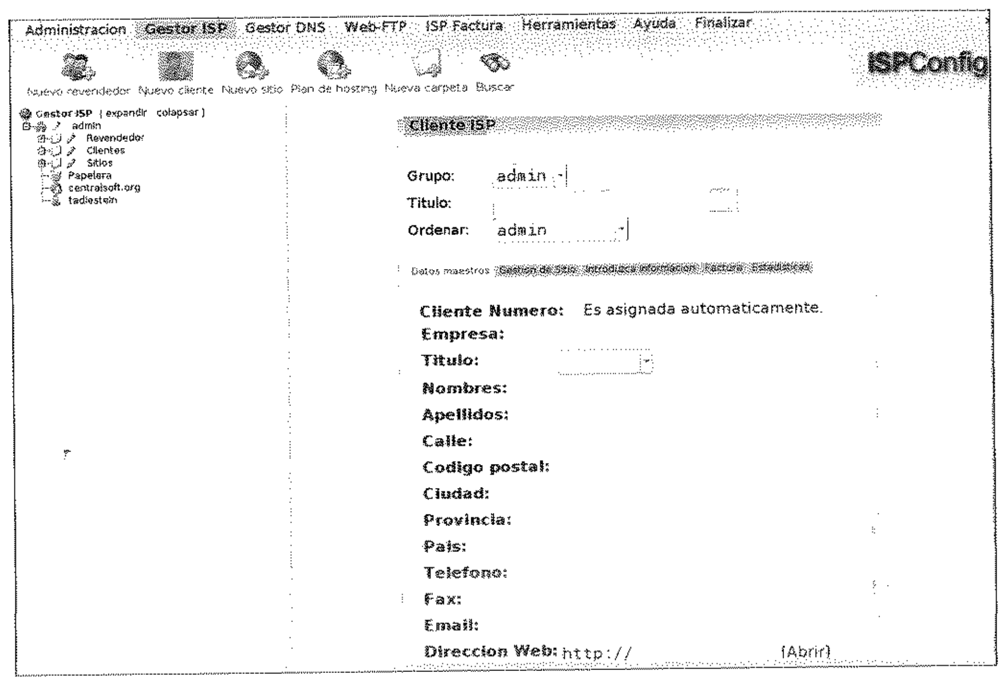


Figura 4.7. Formulario de información del cliente.

Introduzca la información relevante del cliente. La figura 4.8 muestra cómo hemos rellenado el formulario. Fíjese en que hemos usado `Linhelp.org` como el nombre de la compañía.

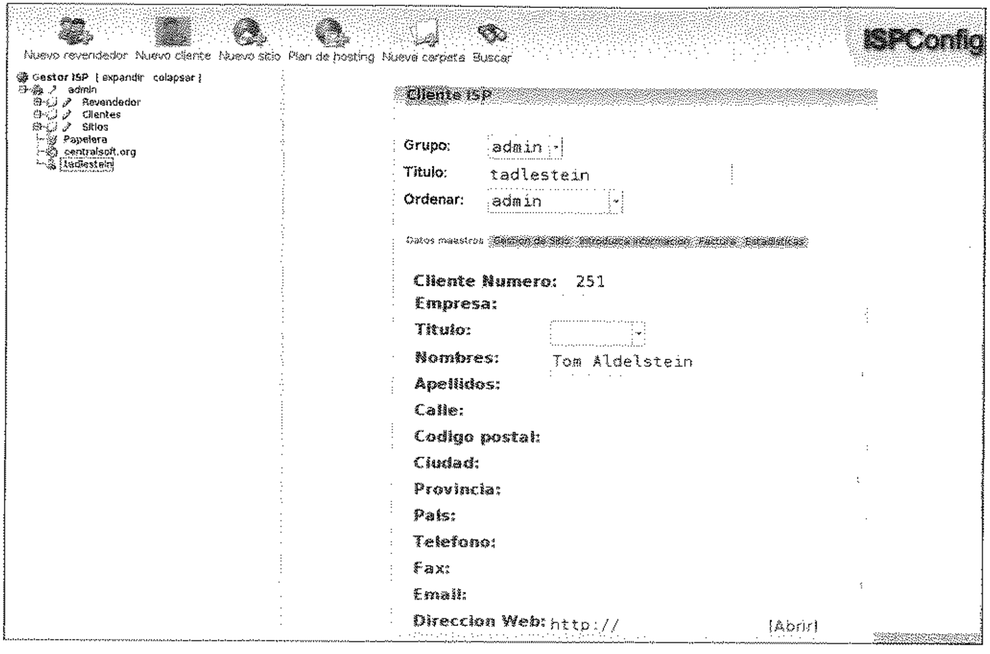


Figura 4.8. Formulario completo para el cliente administrativo.

En la parte izquierda del menú de navegación, verá un nuevo icono representando una persona y acompañado por el nombre del cliente. Ahora ya puede configurar un sitio Web. Simplemente seleccione Nuevo sitio desde la barra de herramientas y verá el cuadro de la figura 4.9.

Proporcione al sitio web un nombre y la dirección IP, y cree un registro DNS. Fíjese también en las solapas del formulario, en la parte de arriba de área donde hay que introducir el nombre:

- Base
- Usuario y correo
- Co-Dominios
- Estadísticas
- Opciones
- Cuenta

Cada una de estas solapas ofrece varias funciones de configuración y gestión. La figura 4.9 no muestra las opciones de la solapa Base. También encontrará algunas otras opciones que le puede dar al administrador de sitio. Para nuestro sitio, hemos ofrecido acceso por línea de comandos, base de datos, creación de FTP y opciones de autentificación, como se muestra en la figura 4.10.

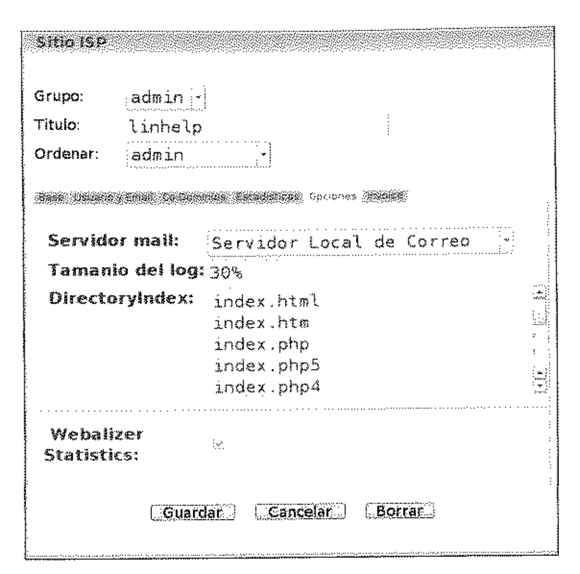


Figura 4.9. El formulario usado para crear el sitio Web linhelp.org.

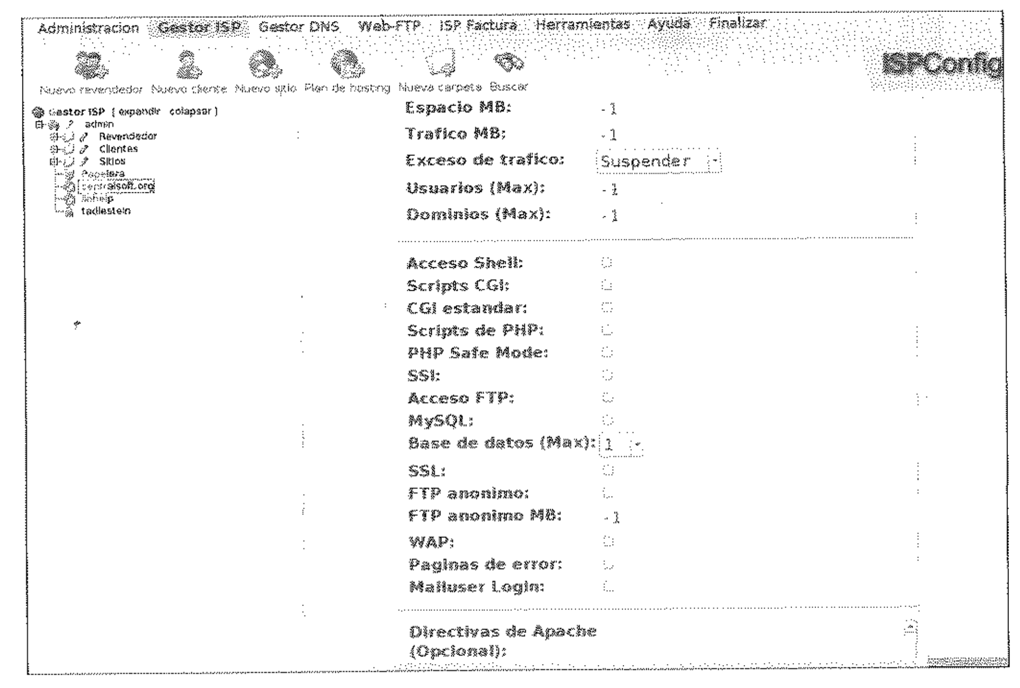


Figure 4.10. Opciones del sitio Web.

Fíjese también en la figura 4.10 debajo de FTP Anónimo, el sistema pone por defecto -1. Esto permite al sitio ofrecer espacio ilimitado para FTP. Esto sería útil si quisiera ofrecer acceso como mirror de un sitio de descargas, por ejemplo. En caso contrario, es mejor definir un límite de manera que nadie pueda subir tantos datos como para saturar el espacio en disco necesario para otros servicios. En este punto, ya tiene un sitio Web útil. En la figura 4.11 se muestra, además, una forma fácil de añadir páginas usando un cliente FTP gráfico como gftp para transferir un sitio que ha sido creado en su escritorio.

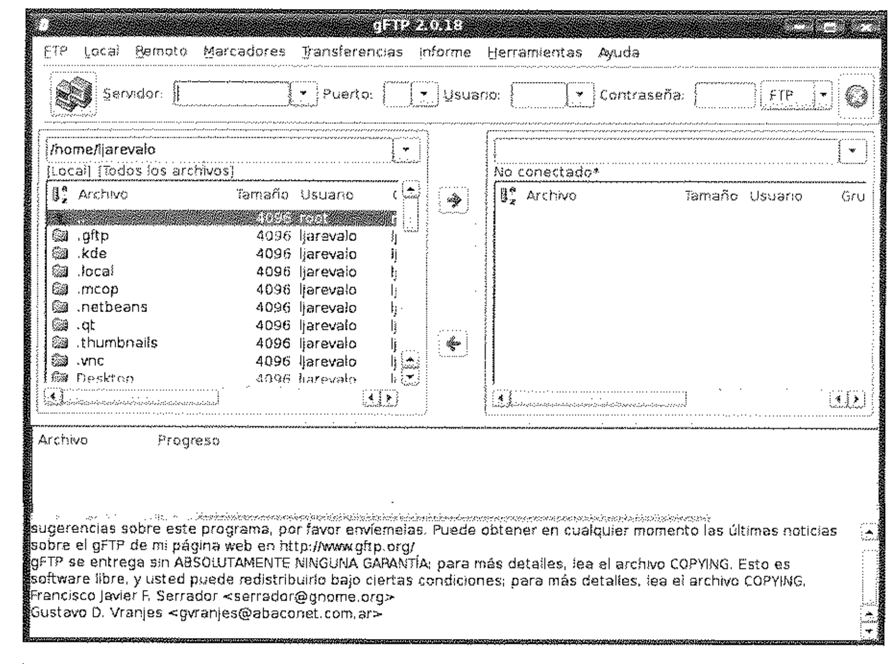


Figura 4.11. Usando gftp para transferir archivos al directorio raíz linhelp.org.

Poniendo en un navegador `http://linhelp.org` ahora podrá ver nuestra página index.html. Puede ver dicha página en la figura 4.12.

Ahora ya tenemos un sitio Web simple y funcional. Echemos un vistazo a la figura 4.13 para comprender lo que estamos configurando.

ISPConfig usa un modelo jerárquico con `/var/www/web1/web` como el directorio raíz para el puerto 80. En cada directorio que cree en esta ruta, Apache crea otra rama donde puede poner páginas. Por defecto, cuando un navegador solicita el directorio. Apache busca el archivo HTML llamado index.html para mostrarlo. Si no ha creado un archivo `index.html`, se mostrarán los nombres de los archivos que están en el directorio raíz.

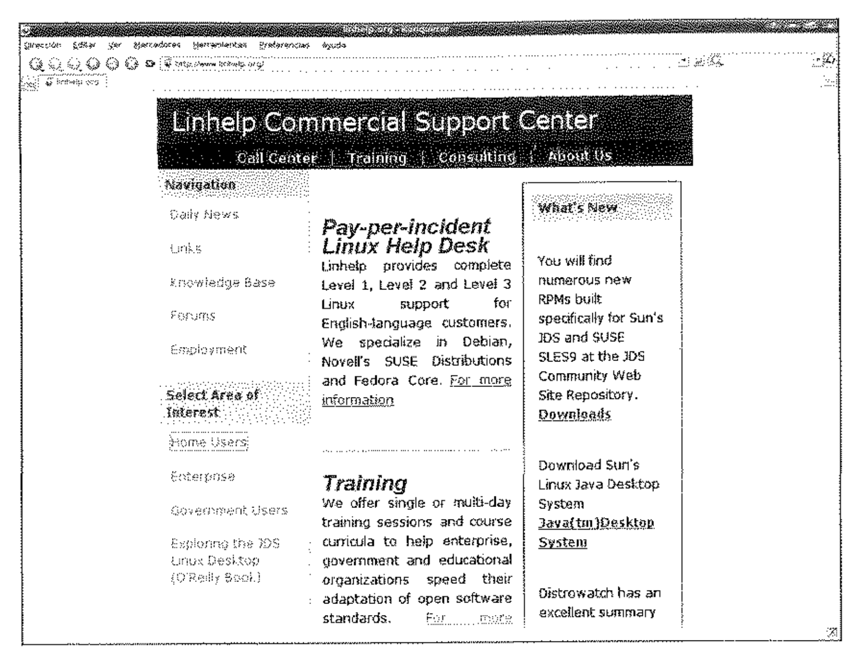


Figura 4.12. El sitio web Linhelp.org.

La figura 4.13 ofrece un ejemplo del directorio raíz de un sitio Web. La página de inicio se muestra siempre que el navegador especifique el nombre del directorio, debido a que tiene el título por defecto. La página de inicio contiene enlaces a otras páginas del sitio.

El diagrama de ejemplo de la figura 4.13. puede tratarse como un gráfico de sectores. El código actual con la página de inicio se parece a esto:

```
<a href="./about_us.html">About Us </a><br><br>
<a href="./products.html">Products </a><br><br>
<a href="./services.html">Services </a><br><br>
<a href="./support.html">Support </a><br><br>
```

Normalmente, el equipo Web al que le dé soporte creará la estructura de directorios y páginas Web. Probablemente necesite ofrecerles una base de datos también, pero esto se tratará en otro capítulo. Por ahora, sólo necesita saber cómo establecer un sitio Web y un dominio de Internet.

Gestionando usuarios y correo electrónico

Una de las tareas más importantes a la hora de administrar sistemas Linux es la gestión de usuarios y sus cuentas. Puede usar el panel gráfico ISPConfg. Una

vez que haya configurado los dominios, al seleccionar uno de ellos en la sección de gestión de ISP de la barra de tareas traerá al primer plano la pantalla del sitio que se mostró en la figura 4.9. Volvamos atrás para verla.

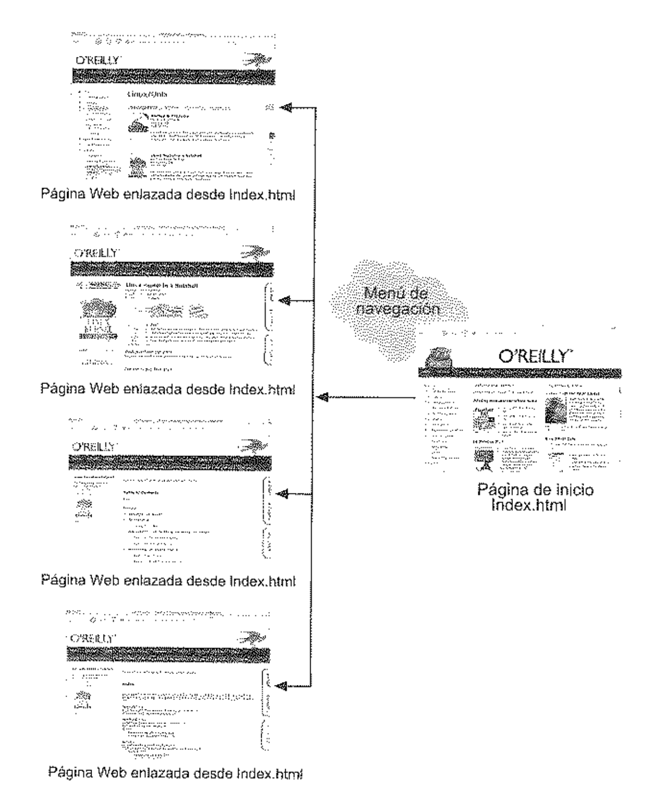


Figura 4.13. Estructura de un sitio web simple.

El formulario tiene seis solapas. La segunda solapa de la izquierda se llama Usuario y Correo. Desde esta solapa podrá añadir nuevos usuarios y gestionar los existentes. Cuando selecciona Nuevo, podrá ver otro formulario como el de la figura 4.14.

En este formulario, puede introducir los detalles de nuevos usuarios y establecer los límites de almacenamiento. Un valor de -1 ofrece espacio ilimitado, pero puede gestionar las cuotas de la manera que prefiera.

En la solapa Opciones Avanzadas (figura 4.15), puede usar una opción para permitir que el correo enviado a un usuario sea redirigido a otra dirección. En otras palabras, si el usuario tiene una dirección de correo alternativa que desea usar, puede usar dicha opción para enviarle el correo a esa cuenta.

Usuario ISP

Nombre Real:

Direccion email: @centralsoft.org

Usuario: web1_

Clave:

Espacio web MB: -1

Espacio mail MB: -1

Administrador:

Guardar Cancelar Borrar

Figure 4.14. Formulario de usuario.

Usuario ISP

Reenviar:

Hint: un email por linea.

Mantener Copia:

Alias de email:

Hint: un alias por linea.

catchAll-Email:

MailScan:

Figura 4.15. Opciones avanzadas de correo.

Otras opciones de esta solapa son:

- Mantener una copia: Al seleccionar esta opción, se mantendrá una copia de cualquier correo electrónico que se reciba en el buzón local del usuario. Esto es muy útil para el caso en que los mensajes que se reciban no hayan llegado a la dirección de correo (debido al filtrado de spam u otro problema).

- Alias de correo: Si no quiere exponer su buzón de correo públicamente, los visitantes del sitio pueden enviar correo a un nombre genérico, tal como `info@centralsoft.org` o `webmaster@centralsoft.org`. Puede hacer esto ofreciendo un alias de correo.
- Coger todo el correo: Esta opción redirige al buzón especificado todos los correos que se hayan enviado a cuentas de usuarios inexistentes. La gente normalmente escribe a direcciones usadas frecuentemente como `editor@centralsoft.org` o `publicidad@centralsoft.org` sin verificar que estas direcciones son válidas. Puede recoger estos mensajes en una cuenta de usuario creada a tal efecto.
- Escanear correo: Si quiere escanear el correo en busca de virus o código Javascript para servidor, use esta opción.
- Autoresponder: Esta opción le permite enviar una respuesta automática a los mensajes enviados a un usuario específico, esto es útil, por ejemplo, cuando el usuario va a estar fuera de la oficina por un período de tiempo.

Con respecto a la solapa Filtro de Spam & Antivirus, mostrada en la figura 4.16, puede considerar qué estrategia de spam usar. Activando el filtro de spam para una cuenta, puede especificar el comportamiento del filtro.

Usuario ISP
Usuario y Email | Configuración avanzada | Spamfilter & Antivirus
Spamfilter:
Antivirus:
Estrategia Spam: aceptar
Spam Hits: 5.0
Rewrite Subject:
Asunto: ***SPAM***
Use URIBL:
Spam Whitelist:
Hint: One entry per line. Valid entries are, for example, friend@somewhere.com, *@isp.com, or *.domain.net

Figura 4.16. La solapa Filtro de spam & Antivirus.

Si selecciona la estrategia de aceptar spam, permite que el spam entre en el buzón del usuario, siendo el agente de correo del usuario (MUA) el encargado de filtrar el spam. Muchos administradores prefieren esta estrategia inicialmente, hasta que el usuario tenga una base de datos de correo identificado como spam. Después, el usuario puede cambiarse al modo de descarte, donde el correo identificado como spam se borra del servidor.

Ahora echemos un vistazo a las opciones para el spam:

- Aciertos de spam: El filtro de spam ejecuta un determinado número de pruebas sobre correos entrantes y asigna puntos a cada prueba. Si la suma de los para estas prueba alcanza o supera el valor especificado por los Aciertos de spam, el correo se clasifica como spam y se maneja de acuerdo con la estrategia definida por el usuario.
- Reescribir asunto: En el modo aceptar, escoger esta opción significa que la línea que identifica el asunto de cada correo se identificará como spam añadiendo un prefijo identificativo (por defecto ***SPAM***). Esto permite al usuario ordenar el correo electrónico según el asunto.

Para permitir que un usuario haga cambios en su cuenta de correo electrónico por sí mismo (incluso la contraseña, el filtrado de spam y las opciones de spam), debe seleccionar la opción Usuario de correo para dicho usuario en la solapa Base del formulario del sitio (Véase la figura 4.10). Para hacer cambios, el usuario de correo puede autentificarse en el sistema a través de la dirección: `http://centralsoft.org:81/mailuser.sci`.

Directorios públicos, de usuario y de inicio

Cada usuario de un dominio gestionado por ISPConfig tiene su propio directorio de inicio en el directorio de usuarios. Si el acceso FTP está permitido, los usuarios aparecerán en su directorio de inicio al autentificarse vía FTP. Cada directorio de inicio también contiene una carpeta llamada Web a la que los usuarios pueden acceder visitando una URL del tipo: `http://www.centralsoft.org/~user` o `http://www.centralsoft.org/users/user`.

La figura 4.17 muestra la estructura de un directorio de inicio para el usuario creado en `centralsoft.org`.

Configuración del cliente de correo electrónico

En este punto, debería comprender los aspectos básicos a la hora de configurar un sitio Web, crear un usuario y manejar el correo. Pero además, tendrá que ser capaz de ayudar a sus usuarios a configurar sus clientes de correo electrónico, especificando los servidores de correo entrante y saliente. En nuestro sistema, ISPConfig usa `server1.centralsoft.org` tanto como servidor SMTP como servidor POP3/IMAP.

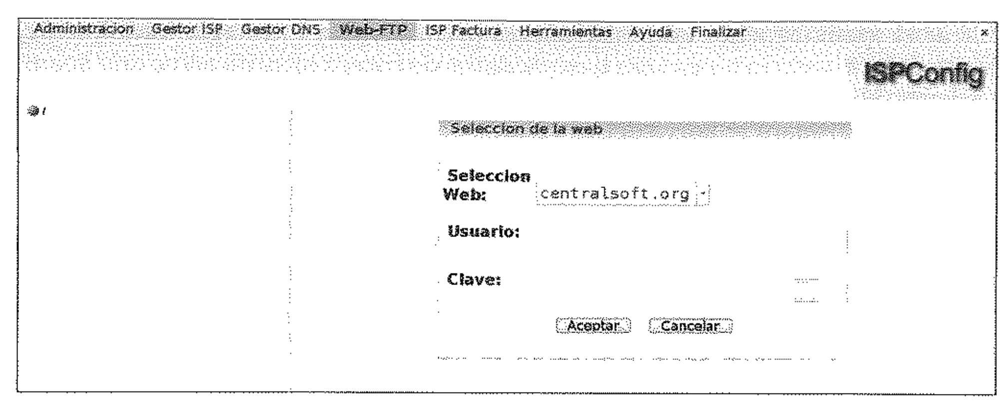


Figura 4.17. Vista de tipo navegador para el directorio de usuario.

En los clientes modernos de correo electrónico, existe la opción de elegir la capa de transporte seguro (TLS). Seleccione TLS cuando sea posible para configurar el servidor de correo saliente. Debido a que la mayoría de los clientes de correo usan su ISP como servidor SMTP, puede seleccionar TLS si su ISP lo usa. En la gran mayoría de casos, su ID de usuario y contraseña viajan sobre las líneas de su ISP en forma de texto plano.

Para recibir correo, configure el servidor de entrada (nosotros usamos `server1. centralsoft.org`) y seleccione o POP3 o IMAP. Use su nombre de sistema (por ejemplo: `web1_adelstein`) y especifique la dirección de correo como el alias (por ejemplo: `tom@centralsoft.org`).

Si aparece un mensaje de error como "-ERR Unknown AUTHORIZATION state command" al intentar obtener el correo vía POP3, probablemente se haya olvidado de activar la encriptación SSL/TLS. Reconfigure su cliente de correo, active POP3-sobre-SSL e inténtelo de nuevo.

Salvaguardando un servidor Web Linux

En el entorno actual de los negocios a veces ocurren cosas inesperadas. Algunas personas escanean direcciones IP en busca de fallos. O usan sofisticados diccionarios de contraseñas para intentar conseguir acceso como root a los servidores, de manera que puedan usarlos como plataformas de spam, virus o gusanos. Las situaciones que los administradores de sistemas tienen que afrontar tienen su raíz en una combinación de factores con precisión o certeza. Por lo tanto, los administradores tienen que aprender a adaptarse rápidamente a las nuevas (y hostiles) situaciones.

Hay dos formas de adaptarse. Primero, si está lo suficientemente concienciado, puede tomar precauciones. Nosotros llamamos a esto anticipación.

Otras veces, no obstante, tendrá que adaptarse a la situación en función del momento, sin tiempo para la preparación. Esto requiere improvisación. Para ser completamente adaptable, hay que ser capaz tanto de anticiparse como de improvisar.

El papel de demonio monitorizador de demonios

No importa lo riguroso que sea a la hora de salvaguardar su servidor de Internet, por alguna extraña combinación de razones, su sistema podría fallar. En un mundo perfecto, podría monitorizar cada servicio y el sistema le alertaría inmediatamente del fallo. Pero, no vivimos en un mundo donde todas nuestras expectativas puedan verse cumplidas.

Imagine que aloja su servidor en un ISP a muchos kilómetros de su base de operaciones. Si este servidor se viene abajo, alguien podría llamar al ISP y conseguir que el personal de servicio vuelva a dejarlo funcionando. La persona encargada del soporte técnico puede que no esté disponible inmediatamente, por lo tanto tendrá que esperar un tiempo con una aplicación crítica caída.

En una empresa grande, podría sentirse aislado si su servidor está situado a muchos kilómetros. Los operadores de los centros de datos no suelen conceder acceso a la sala de ordenadores, ni siquiera a los administradores de sistemas independientemente de su ubicación, por lo que es muy importante que un administrador sepa gestionar sus sistemas remotamente.

Un demonio monitorizador de demonios (DMD) es una utilidad que observa sus servicios y automáticamente intenta reiniciarlos cuando fallan. Si un servicio falla, normalmente tiene que autentificarse en el servidor y abrir una consola para ejecutar un comando como `/etc/init.d/mysql restart`. Un DMD puede, sin embargo, ejecutar este comando sin intervención por su parte.

Si el servicio se reinicia, fin del problema. Si no se reinicia satisfactoriamente, el DMD hará un determinado número de intentos (por ejemplo 5) y luego contactará con usted vía mensaje de texto, correo electrónico u otra forma de comunicación que pueda alertarle del problema. En este punto, tendrá que intervenir para averiguar por qué el servicio ha fallado. El DMD se ejecuta como otro servicio de su servidor. Tiene un archivo de configuración para elegir la opción que mejor se ajusta a sus necesidades. Puede hacer que se inicie manual o automáticamente.

En la siguiente sección, configuraremos un DMD llamado monit, que tiene una interfaz Web sencilla como se muestra en la figura 4.18.

Fíjese bien en que hay cinco servicios bajo vigilancia. En la figura 4.19, se puede ver claramente cómo el sistema gestiona cada proceso. En este caso, estamos mostrando sshd.

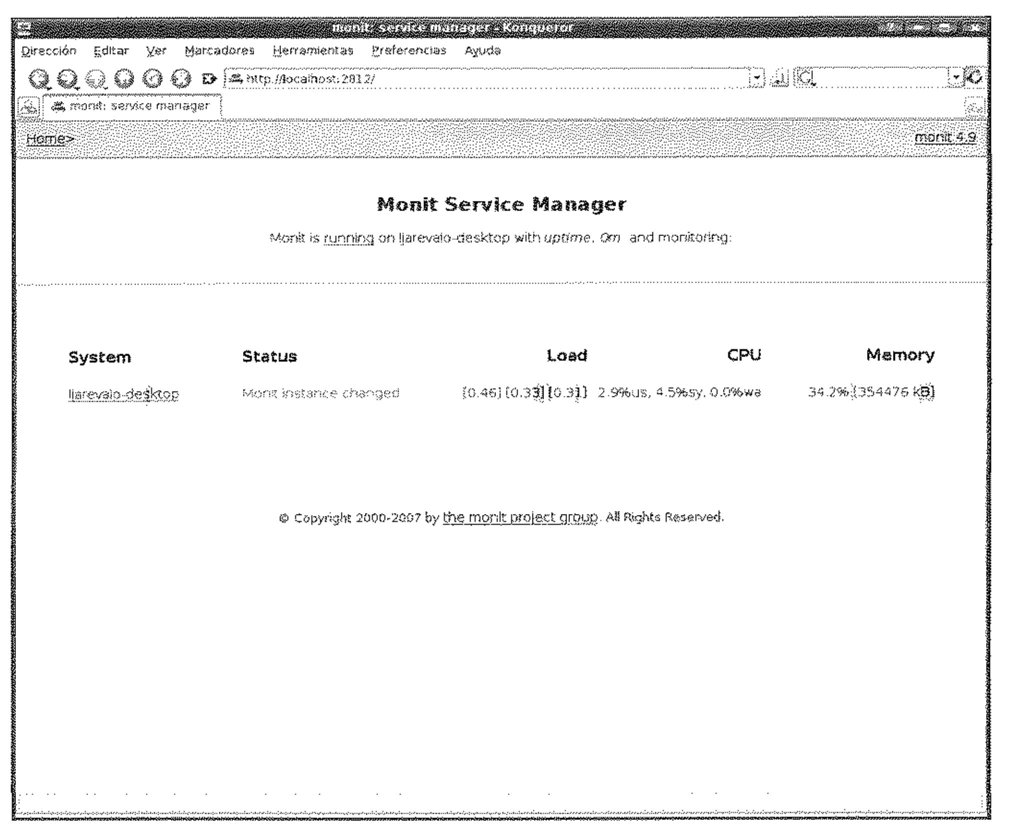


Figura 4.18. Interfaz Web para monit ejecutándose en centralsoft.org.

Fíjese que en la figura 4.19 el estado de sshd muestra que se está ejecutando y que el sistema lo está monitorizando. En las tres líneas de la parte de debajo de la pantalla, puede ver las instrucciones que se harán si sshd falla:

```
If failed localhost:22 [SSH] with timeout 5 seconds then restart else
if recovered
then alert
```

Esta política simplemente reinicia un servicio que ha fallado y envía un mensaje cuando se reinicia satisfactoriamente.

Finalmente, monit ofrece cuatro botones en la parte de debajo de la página para la intervención manual. Ahora, veamos cómo funciona el sistema.

Instalando y configurando monit

Para poder instalar monit tiene dos opciones igual de válidas. Puede utilizar tanto el gestor de paquetes del sistema como también descargar la distribución desde `http://www.tildeslash.com/monit`. Si está usando Debian, simplemente introduzca:


```
# apt-get install monit
```

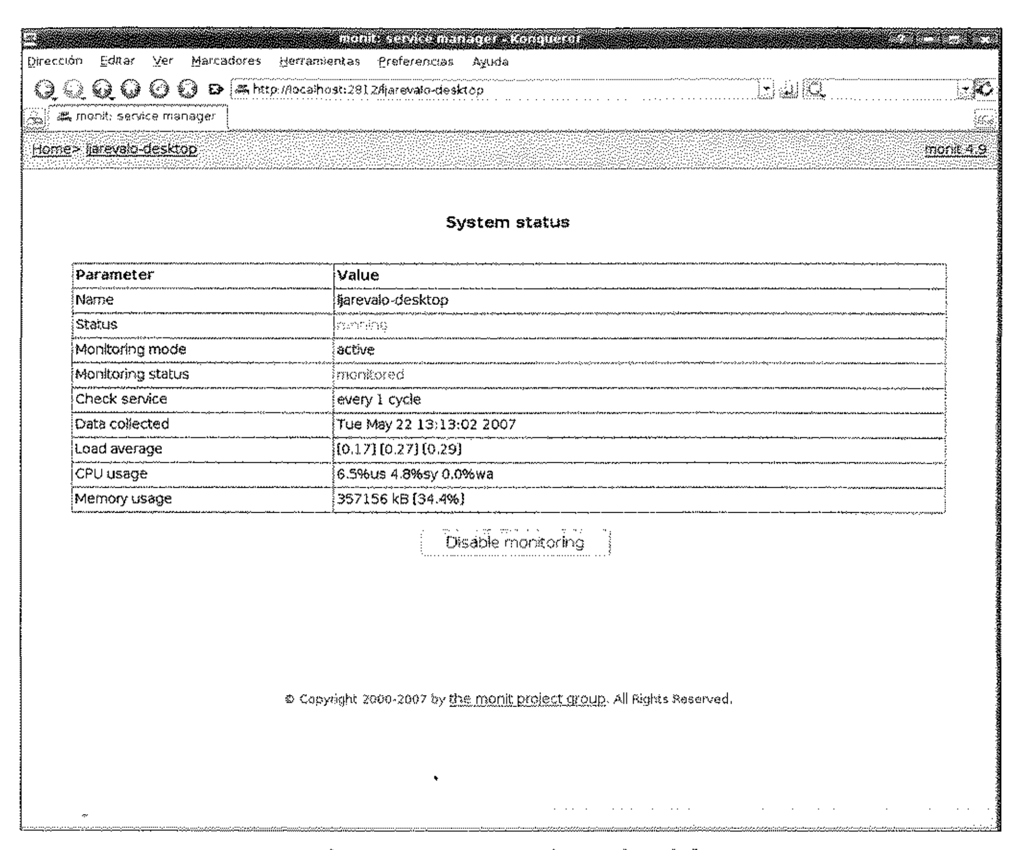


Figura 4.19. Inspeccionando sshd.

Después de que haya instalado monit, edite `/etc/monit/monitrc`. El archivo creado durante la instalación contiene montones de ejemplos y puede encontrar más ejemplos de configuración en `http://www.tildeslash.com/monit/doc/examples.php`. En nuestro caso, nosotros queremos:

- Activar la interfaz Web de monit en el Puerto 2812.
- Monitorizar los servicios proftpd, sshd, mysql, apache y postfix.
- Crear una interfaz Web basada en Secure Sockets Layer (https) donde poder autentificarnos como admin.
- Indicarle a monit que manda alertas de correo a `root@localhost`.

Nuestro archivo de configuración /etc/monit/monitrc es:

```
set daemon 60
set log file syslog facility log_daemon
set mailserver localhost
set mail-format { from: monit@server1.centralsoft.org }
set alert root@localhost
set httpd port 2812 and
      SSL ENABLE
      PEMFILE /var/certs/monit.pem
      allow admin: test
check process proftpd with pidfile /var/run/proftpd.pid
   start program = "/etc/init.d/proftpd start"
   stop program = "/etc/init.d/proftpd stop"
   if failed port 21 protocol ftp then restart
   if 5 restarts within 5 cycles then timeout
check process sshd with pidfile /var/run/sshd.pid
   start program "/etc/init.d/ssh start"
   stop program "/etc/init.d/ssh stop"
   if failed port 22 protocol ssh then restart
   if 5 restarts within 5 cycles then timeout
check process mysql with pidfile /var/run/mysqld/mysqld.pid
   group database
   start program = "/etc/init.d/mysql start"
   stop program = "/etc/init.d/mysql stop"
   if failed host 127.0.0.1 port 3306 then restart
   if 5 restarts within 5 cycles then timeout
check process apache with pidfile /var/run/apache2.pid
   group www
   start program = "/etc/init.d/apache2 start"
   stop program = "/etc/init.d/apache2 stop"
   if failed host www.centralsoft.org port 80 protocol http
      and request "/monit/token" then restart
   if cpu is greater than 60% for 2 cycles then alert
   if cpu > 80% for 5 cycles then restart
   if totalmem > 500 MB for 5 cycles then restart
   if children > 250 then restart
   if loadavg(5min) greater than 10 for 8 cycles then stop
   if 3 restarts within 5 cycles then timeout
check process postfix with pidfile /var/spool/postfix/pid/master.pid
   group mail
   start program = "/etc/init.d/postfix start"
   stop program = "/etc/init.d/postfix stop"
   if failed port 25 protocol smtp then restart
   if 5 restarts within 5 cycles then timeout
```

Las sentencias y las opciones se describen en la documentación de monit en http://www.tildeslash.com/monit/doc/manual.php.

En la sección de apache de la configuración de monit, verá la sentencia:

```
if failed host www.centralsoft.org port 80 protocol http
   and request "/monit/token" then restart
```


Esto significa que monit intenta conectarse con www.centralsoft.org en el Puerto 80 e intenta acceder al archivo /monit/token. Debido a que el documento raíz del sitio Web está en /var/www/www. centralsoft.org/web, el nombre del archivo se extiende a /var/www/www.centralsoft.org/web/monit/token. Si monit no se ejecuta, significa que Apache no se está ejecutando, por lo que monit intentará reiniciarlo.

Ahora debemos crear el archivo var/www/www.centralsoft.org/web/monit/token y escribir una cadena de texto arbitraria:

```
# mkdir /var/www/www.centralsoft.org/web/monit
# echo "hello" > /var/www/www.centralsoft.org/web/monit/token
```

Puede seguir un procedimiento similar en su sistema.

Luego, cree un directorio para albergar el archivo de certificado (/var/certs/monit.pem) necesario para interfaz Web SSL de monit:

```
# mkdir /var/certs
# cd /var/certs
```

Necesitará un archivo de configuración OpenSSL para crear el certificado. El resultado /var/certs/monit.pem debería ser como este:

```
# create RSA certs - Server
RANDFILE = ./openssl.rnd
[ req ]
default_bits = 1024
encrypt_key = yes
distinguished_name = req_dn
x509_extensions = cert_type
[ req_dn ]
countryName = Country Name (2 letter code)
countryName_default = MO
stateOrProvinceName              = State or Province Name (full name)
stateOrProvinceName_default      = Monitoria
localityName                     = Locality Name (eg, city)
localityName_default             = Monittown
organizationName                 = Organization Name (eg, company)
organizationName_default         = Monit Inc.
organizationalUnitName           = Organizational Unit Name (eg, section)
organizationalUnitName_default   = Dept. of Monitoring Technologies
commonName                       = Common Name (FQDN of your server)
commonName_default               = server.monit.mo
emailAddress                     = Email Address
emailAddress_default             = root@monit.mo
[ cert_type ]
nsCertType = server
```

Ahora cree el certificado:

```
# openssl req -new -x509 -days 365 -nodes -config ./monit.cnf -out \
/var/certs/monit.pem -keyout /var/certs/monit.pem
```

```
# openssl gendh 512 >> /var/certs/monit.pem
# openssl x509 -subject -dates -fingerprint -noout -in /var/certs/
monit.pem
# chmod 700 /var/certs/monit.pem
```

Luego edite /etc/default/monit para activar el demonio monit. Cambie startup a 1 y configure CHECK_INTERVALS con el intervalo en segundos que quiere que se compruebe el sistema. Nosotros hemos elegido 60. El archivo debería quedar así:

```
# Defaults for monit initscript
# sourced by /etc/init.d/monit
# installed at /etc/default/monit by maintainer scripts
# Fredrik Steen <stone@debian.org>
# You must set this variable to for monit to start
startup=1
# To change the intervals which monit should run uncomment
# and change this variable.
CHECK_INTERVALS=60
```

Finalmente, inicie monit:

```
# /etc/init.d/monit start
```

Ahora haga que su navegador apunte a https://your_domain:2812/ (asegúrese de que el puerto 2812 no está bloqueado por su cortafuegos) y autentifíquese con el usuario admin y la contraseña test. Debería ver la interfaz Web de monit, tal y como se mostró anteriormente en la figura 4.18.

Qué es lo próximo

Empezamos levantando el servidor y configurándolo de manera que pueda usarse como una plataforma para Internet. Hemos instalado un servidor basado en texto sin el sistema de X Windows (por razones de seguridad y de rendimiento) y luego hemos configurado interfaces basadas en Web que le permitan gestionar de manera segura y monitorizar sus servicios.

En los restantes capítulos, vamos a profundizar nuestra exploración del sistema de administración de Linux. En el capítulo siguiente, aprenderá a instalar software administrativo que no se instala automáticamente. Configuraremos la mayoría de las aplicaciones Linux que la gente usa cada día en la empresa y en negocios de pequeño y medio tamaño.

Capítulo 5

Correo



Este capítulo muestra cómo levantar un servicio de correo electrónico para un sitio de pequeño o medio tamaño. Los elementos del servicio son:

- El servidor Postfix como agente de transferencia de correo SMTP (MTA), que acepta correo de otros usuarios e interactúa con otros sitios a través de Internet para enviar el correo.
- Servidores Post Office Protocol (POP) y el Protocolo interactivo de acceso al correo (IMAP) para entregar el correo a sus usuarios.
- La capa *Simple Authentication and Security Layer* (SASL) sirve para autentificar el correo y evitar el spoofing.

Configuraremos Postfix para usar el tradicional sistema de autentificación basado en archivos, que puede soportar miles de usuarios. Las grandes instalaciones de correo pueden almacenar nombres de cuentas de correo y contraseñas en bases de datos relacionales o en directorios LDAP. Para ver un ejemplo de un servidor de correo muy escalable basado en Postfix con autentificación LDAP, véase Zimbra (http://www.zimbra.com).

Las soluciones de este capítulo intentan juntar diversos componentes para hacer un sistema de gestión de correo eficiente, robusto y seguro. Hoy en día, personas como Wietse Venema (el inventor de Postfix) han conseguido reducir mucho la complejidad de configuración de los sistemas de correo electrónico. En lugar de vérselas con complejas configuraciones de servidores de correo, los administradores de sistemas Linux tienen otros problemas más importantes que resolver:

- Cómo asegurar el correo, una forma de comunicación que no fue diseñada teniendo en cuenta la seguridad, ni siquiera contra los ataques de suplantación de identidad u otros ataques maliciosos.

- Cómo proteger datos sensibles para la compañía.
- Cómo dar acceso a usuarios que están fuera de la red de la compañía.

Aspectos claves del servicio de correo

Los agentes de transferencia hacen la parte más dura de la comunicación en Internet, moviendo el correo de un sitio para otro en Internet. Para enviar un correo electrónico, el remitente liga su sistema a un MTA, que luego usa SMTP para transferir el correo al MTA responsable de entregar dicho correo a su destinatario.

El destinatario tiene distintas formas de recuperar el correo desde un MTA, ninguna de ellas usa SMTO: puede autentificarse como usuario del sistema y ejecutar el MTA, indicarle al MTA una conexión directa (al igual que se le indica al ISP una línea telefónica) o a través de un túnel en Internet desde un MTA remoto. (Estamos ignorando otros métodos como hacerlo desde una interfaz Web como Gmail o usando un teléfono móvil.) Independientemente del método que el receptor use, este recupera su correo a través de un agente de entrega de correo (MDA) como Courier IMAP. El MDA habla con el MTA para obtener el correo y ofrece un buzón para almacenar el correo. El correo puede mostrarse al usuario a través de un agente de usuario de correo (MUA), como Outlook, Evolution o Thunderbird.

Los usuarios normalmente recuperan su correo usando o POP3 o IMAP4 sobre TCP/IP. Casi todos los MUA modernos soportan tanto POP3 como IMAP4. Los MUA envían el correo pasándoselo a una MTA y transfiriéndolo sobre SMTP.

La mayoría de la gente mantiene listados de las direcciones de sus contactos en su MUA, de modo que puedan buscar la dirección de correo de la gente. En entornos empresariales, los contactos se suelen almacenar en servidores de directorio LDAP. Un montón de usuarios ignoran que pueden encontrar sus contactos en los sistemas LDAP.

Postfix, Sendmail y otros MTA

Quizá se pregunte por qué hemos escogido Postfix como MTA en lugar de Sendmail, el servidor de correo para Internet original y que fue desarrollado a principios de la década 1980 por Eric Allman en la Universidad de California-Berkeley. Sendmail estuvo presente en la mayoría de las instalaciones de MTA en Internet, pero no estamos seguros de que esto siga siendo así. Muchos sondeos indican que la popularidad de Sendmail ha caído rápidamente, hasta llegar al 40

por 100 de los servidores de Internet. Aunque los defensores de Sendmail defienden que es flexible y escalable, muchos administradores de sistemas lo consideran extremadamente complejo y difícil de configurar y mantener.

Sendmail se desarrolló antes que aparecieran el spam y el software dañino, por lo tanto tiene algunas debilidades en cuanto a seguridad. Uno de los problemas más serios de Sendmail es que, por defecto, permite que la pasarela quede abierta, es decir, que procesa cualquier correo originado por un extraño. Este problema de seguridad está ilustrado en la figura 5.1.

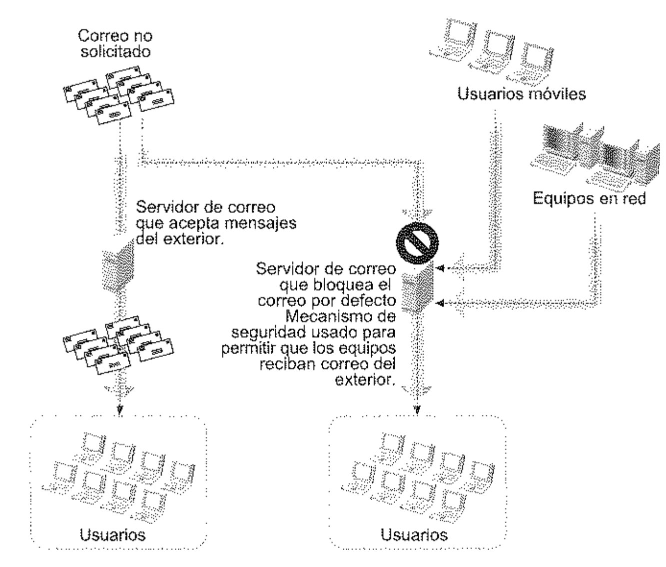


Figura 5.1. Aspectos de seguridad en un entorno hostil para el correo.

Remitentes de correo no solicitado (UBE), también conocido como spammers o remitentes de correo no deseado (UCE) son los responsables de más del 50 por 100 del tráfico de correo de Internet. Esto provoca que las colas de correo, los servidores DNS, la capacidad de procesamiento y de almacenamiento y las infraestructuras se resientan. Los UBE tienen una variedad de técnicas que les permite esconder su identidad real, incluyendo la suplantación de direcciones IP, la creación de correo basura y su dispersión a través de los servidores SMTP abiertos.

Los MTA bien configurados aceptan correo saliente sólo para las direcciones que pertenecen a sus usuarios legítimos, esto está normalmente limitado a una subred determinada. Pero, por defecto, Sendmail procesa el correo de cualquiera. Si usa Sendmail y no deshabilita esta opción, los UBE pueden usar su MTA para ocultar su origen. Su servidor de correo podría estar en listas negras para pasare-

las abiertas, lo que podría provocar que correo legítimo fuera tratado como spam. En teoría, puede tener problemas legales si facilita la distribución de correo basura.

La base de usuarios de Sendmail a menudo opera con versiones obsoletas, sin parchear y sin documentar, por lo que facilitan las cosas a los UBE. Los desarrolladores de Sendmail son conscientes del problema y trabajan duro para hacerlo más seguro, pero los mejores avances de seguridad se consiguen con cada versión de pago del producto. Aaron Weiss aclara algunas dudas entre la versión gratuita y la versión de pago de Sendmail en el artículo "The Fee vs. Free Divide" (`http://www.serverwatch.com/tutorials/article.php/3580006`):

Para comercializar Sendmail y ofrecer productos de valor añadido que mejor el desarrollo, se creó Sendmail Inc. Su producto estrella, Sendmail Switch está basado en la versión libre de Sendmail. Se asienta en su núcleo e incluye una consola gráfica de gestión, facilidades para la seguridad, soporte técnico, filtros de gestión de contenidos (incluyendo defensas anti-spam y antivirus), soporte para SSL, SASL y directorios LDAP, y capacidades de auditoría, clustering y gestión remota. Todo ello acompañado de un instalador gráfico y asistentes para tareas.

Además, el Sendmail Consortium (responsable de la versión libre del MTA Sendmail) está patrocinado por Sendmail Inc. Ofrece módulos anti-spam, antivirus y para la gestión remota del MTA. Lo siguiente es una descripción del modelo de negocio de Sendmail Inc (obtenida en `http://www.sendmail.com/company`):

Sendmail ofrece diferentes soluciones empresariales para mensajería segura entre las que se incluye correo electrónico, voz y mensajería instantánea. Las soluciones Sendmail controlan el correo entrante, saliente e interno. Sendmail es una implementación portable, que puede desplegarse sobre gran variedad de software y aplicaciones. Los productos de Sendmail funcionan en las infraestructuras de correo heterogéneas como Exchange, Notes, Groupwise y otras soluciones de correo.

Postfix se diseñó desde el principio como un sustituto robusto y seguro para Sendmail. El MTA por defecto para Debian es Exim 4, pero preferimos Postfix porque Exim tiene algunos problemas de escalabilidad. Carece de una cola central de gestión y capacidad de carga balanceada. Además, existen indicios de que los desarrolladores de la distribución Debian van a convertir a Postfix en el MTA por defecto en un futuro cercano. Al mismo tiempo, es fácil migrar de Exim a Postfix, como veremos en la siguiente sección.

El servidor SMTP de correo Postfix en Debian

Para levantar nuestro servidor, vamos a hacer uso de una instalación reciente de Debian. Si elige otra distribución, puede hacer lo mismo que lo que indicamos y conseguirá los mismos resultados.

Paquetes de Debian relacionados con Postfix

Use la última versión estable de Debian y configúrela con el número mínimo de paquetes. Si todavía no tiene un disco de instalación en red para Debian, descárguelo de `http://www.us.debian.org/CD/netinst`. Después haga una instalación de red y asegúrese de que proporciona un nombre de dominio correcto. Luego, configure Debian como se sugiere en esta sección.

El instalador de Debian le conduce por un script estándar antes de la configuración, siga la rutina de instalación estándar hasta que tenga que elegir el tipo de instalación que desea. La pantalla que podrá ver será así:

```
( ) Entorno de escritorio
( ) Servidor Web
( ) Servidor de Impresión
( ) Servidor DNS
( ) Servidor de archivos
( ) Servidor de correo
( ) Base de datos SQL
( ) Selección manual de paquetes
```

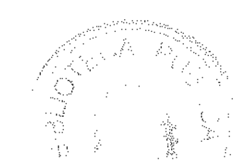
No seleccione ninguna opción, no va a usar el servidor de correo por defecto (Exim) puesto que en su lugar va a instalar Postfix. Sólo presione el tabulador y haga clic en el botón **OK**. El instalador de Debian procederá a descargar y a instalar los paquetes. Durante las descargas, mostrará una o más pantallas gráficas donde preguntará si quiere configurar Exim (Exim-config). Elija Sin configuración. Luego responda Sí cuando le pregunte ¿De verdad que desea dejar el sistema de correo sin configurar?.

El instalador de Debian continuará descargando y configurando paquetes, cuando Debian acabe su trabajo, verá una pantalla agradeciendo el uso de Debian. En este punto debería eliminar algunos programas innecesarios usando la utilidad apt-get. Si decide usar otra distribución, puede eliminar los paquetes usando otros procedimientos. Bajo Debian, ejecute:

```
# apt-get remove lpr nfs-common portmap pidentd pcmcia-cs pppoe \
pppoeconf ppp pppconfig
```

Ahora, deshabilite algunos scripts de servicio:

```
# update-inetd --remove daytime
# update-inetd --remove telnet
# update-inetd --remove time
# update-inetd --remove finger
# update-inetd --remove talk
# update-inetd --remove ntalk
```



```
# update-inetd --remove ftp
# update-inetd --remove discard
```

y ahora reinicie el superservidor inetd:

```
# /etc/init.d/inetd reload
```

Instalando Postfix en Debian

El siguiente comando instala los paquetes necesarios para ejecutar Postfix, junto con la seguridad TLS y SASL que permite autentificar a los usuarios:

```
# apt-get install postfix postfix-doc postfix-tls libsasl2 \
sasl2-bin libsasl2-modules
```

Cuando instala estos paquetes, Debian debe elegir entre instalar libldap2 al mismo tiempo. Libsasl2 ya debería estar instalado en el sistema.

En este punto, la utilidad de instalación de Debian empezará a descargar y configurar varios archivos. Fíjese en que le aparecerá un diálogo durante el proceso que empezará con las siguientes líneas:

```
Reading Package Lists... Done
Building Dependency Tree... Done
```

Luego, verá una pantalla con mensajes que empieza así:

```
You have several choices for general configuration at this point...
```

En la parte de abajo de la pantalla encontrará una pregunta que nos interesa:

```
General type of configuration?
No configuration
Internet Site
Internet with smarthost
Satellite system
Local only

<Ok>      <Cancel>
```

Elija "Internet Site" incluso si planea usar Postfix para correo local.

Luego, un diálogo de información le indicará que el proceso de instalación está escribiendo el archivo de configuración de Postfix. Si ya tiene un servidor en producción usando Sendmail, ya tendrá un archivo de alias. En este capítulo supondremos que está empezando desde cero, por lo que introduciremos NONE en la siguiente pantalla.

```
The user root (and any other users with a uid of 0) must have mail
redirected via an alias, or their mail may be delivered to /var/mail/nobody.
```



```
This is by design: mail is not delivered to external delivery agents as root.
If you already have a /etc/aliases file, then you possibly need to add this
entry. (I will only add it if I am creating a new /etc/aliases.)
What address should I add to /etc/aliases, if I create the file?
(Enter NONE to not add one.)
Where should mail for root go
NONE
                         <Ok>      <Cancel>
```

La próxima pregunta durante la instalación está relacionada con FQDN. Postfix necesita que el comando hostname devuelva un FQDN como `mail.centralsoft.org`. Pero por defecto, hostname en Debian sólo devuelve mail. Para permitirle configurar el FQDN, el script de instalación ofrece el siguiente diálogo:

```
Your 'mail name' is the hostname portion of the address to be shown on
outgoing news and mail messages (following the username and @ sign).
This name will be used by other programs besides Postfix; it should be the
single, full domain name (FQDN) from which mail will appear to originate.
Mail name?
mail.centralsoft.org
                         <Ok>      <Cancel>
```

Responda <Ok> para aceptar el valor por defecto que aparece en el cuadro de texto azul.

El siguiente diálogo muestra los valores por defecto de los dominios de su servidor:

```
Give a comma-separated list of domains that this machine should
consider itself the final destination for. If this is a mail domain
gateway, you probably want to include the top-level domain.
Other destinations to accept mail for? (blank for none)
server2.centralsoft.org, localhost.centralsoft.org, , localhost
                         <Ok>      <Cancel>
```

Lós dominios listados aparecerán en su archivo de configuración `main.cf`.

La pregunta final está relacionada con los sistemas que tienen sistemas de archivos que no son un estándar:

```
If synchronous updates are forced, then mail is processed more slowly.
If not forced, then there is a remote chance of losing some mail if the
system crashes at an inopportune time, and you are not using a journaled
filesystem (such as ext3).
The default is "off".
Force synchronous updates on mail queue?
                         <Yes>     <No>
```

Debido a que casi todas las distribuciones usan el sistema de archivos ext3 por defecto, puede responder <No> aquí.

En este punto, la instalación finaliza y escribe el archivo de configuración de Postfix. Los parámetros y los valores imprimidos no tendrán sentido para usted por el momento, pero tendrá que ser capaz de encontrarlos en el archivo de configuración y cambiarlos si fuera necesario.

Configuración básica de Postfix

El siguiente archivo es un archivo con la configuración mínima de Postfix, /etc/postfix/main.cf:

```
smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
append_dot_mydomain = no
myhostname =
mydomain =
myorigin = $mydomain
inet_interfaces =
mydestination = $mydomain, localhost.$mydomain, localhost
mynetworks = 127.0.0.0/8
```

Si estuviera configurando Postfix a mano, tendría que rellenar muchos de estos valores manualmente. Por ello es de apreciar el trabajo que ahorra el proceso de instalación de Debian.

Postfix usa una sintaxis simple en la que cada línea consiste en un parámetro de configuración seguido por un signo igual y un valor. Una vez que el parámetro está definido, las líneas posteriores del archivo pueden referirse al parámetro siempre que tengan como prefijo el símbolo del dólar:

```
mydomain = centralsoft.org
myorigin = $mydomain
```

Este ejemplo asigna el valor `centralsoft.org` tanto al parámetro mydomain como a myorigin

Un archivo básico de configuración realiza reparto local únicamente. Espera que haya receptores de correo que tengan cuentas y directorios locales en el servidor de correo. No necesita que el sistema añada el sufijo @ (que debería especificarse con el parámetro append_dot_mydomain). Esto es por lo que el proceso de instalación de Debian también pregunta por dominios, nombres de equipo y dirección de destino.

El gestor de paquetes de Debian configura muchos parámetros `/etc/postfix/main.cf` por usted. La tabla 5.1 muestra las líneas maestras. Una lista completa de parámetros puede encontrarse en el sistema archivo de ejemplo `/usr/share/postfix/main.cf.dist`.

Tabla 5.1. Parámetros claves para la configuración de Postfix.

Parámetro	Explicación
`.smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)`	Especifica el texto que identifica este servidor cuando se está comunicando mediante SMTP con otro servidor. El uso de texto como estandarte es obligatorio según todas las especificaciones SMTP.
`biff = no`	biff es un pequeño proceso Postfix que puede notificar a los usuarios locales que tienen correo Nuevo. Si no tiene usuarios locales, debería desactivarlo. Por defecto en Debian está desactivado.
`append_dot_mydomain = no`	En un entorno como el nuestro, añadir el nombre de dominio a una dirección de correo es tarea del MUA. Este valor significa que Postfix no añade el sufijo @centralsoft.org.
`#delay_warning_time = 4h`	Quite el comentario de esta línea para generar mensajes del tipo "correo retrasado". Nosotros no activaremos esta opción porque empezaremos con un volumen bajo de usuarios y no se esperan retrasos.
`myhostname = server2.centralsoft.org`	Especifique el nombre de equipo de Internet de este sistema de correo. Por defecto se suele usar el nombre de dominio cualificado. $myhostname se usa como valor por defecto para otros muchos parámetros de configuración.
`alias_maps = hash:/etc/aliases alias_database = hash:/etc/aliases`	Especifica el alias de la base de datos usada por el agente local de entrega. Un alias es simplemente un nombre alternativo que alguien usa en lugar del original. Por ejemplo, debería especificar el alias admin para root. Los papeles de estos dos parámetros no son importantes para comprender este capítulo, sólo tenga en cuenta que Postfix mantiene una lista con todos los alias en un archivo simple y que estos parámetros le indican al sistema dónde está situado y cuál es el formato del archivo de datos usado.

Parámetro	Explicación
`myorigin = mydomain`	Especifica el dominio que aparecerá en los correos enviados.
`mydestination = server2.centralsoft.org, localhost.centralsoft.org, localhost`	Especifica una lista de equipos y nombres de dominio, separados por comas o espacios en blanco, para los cuales este servidor aceptará correo.
`relayhost =`	Especifica un equipo por defecto que este servidor usará para devolver correo cuando no sepa localizar al receptor. Nosotros lo hemos dejados en blanco, confiando sólo en el parámetro mynetworks.
`mynetworks = 127.0.0.0/8`	Especifica los equipos que este servidor no considera spammers. Aquí, hemos especificado solamente nuestro equipo local. En su lugar, también puede especificar el parámetro mynetworks_style = clase donde Postfix debería confiar en todos los clientes SMTP pertenecientes a la misma clase de red (A/B/C) que la máquina local. No confíe la clase entera en un servidor de acceso telefónico, ya que esto provocará que Postfix esté abierto para todos los proveedores de la red.
`mailbox_command = procmail -a "$EXTENSION"`	Especifica el comando externo que se usa para entregar el correo en el buzón del usuario. Este comando se ejecuta cuando el receptor tenga definidas las variables de entorno HOME, SHELL y LOGNAME.
`mailbox_size_limit = 0`	Define una cuota para el correo almacenado por cada usuario. 0 desactiva la cuota.
`recipient_delimiter = +`	Especifica el separador usado entre los nombres de usuario y las extensiones de dirección en una tabla de búsqueda.
`inet_interfaces = all`	Especifica la interfaz de red (tarjeta de red) a través de la cual el sistema recibe el correo. Esto es muy útil sólo si tiene más de una tarjeta de red.

Puede que necesite algunas personalizaciones sencillas pero útiles, he aquí algunas de ellas:

- Normalmente, mydestination lista los dominios que aparecen en las direcciones de correo de los usuarios locales, es decir, los dominios para los cuales Postfix acepta y procesa el correo. Por defecto, Postfix acepta correo destinado a $myhostname y a localhost.$mydomain, el equipo en el que Postfix se está ejecutando. Puede especificar que el sistema acepte su dominio entero añadiendo $mydomain a la lista:

```
mydestination = $myhostname, localhost.$mydomain, $mydomain
```

- Puede indicarle a Postfix qué equipos quiere que procesen el correo configurando el parámetro mynetworks. (Si establece mynetworks, Postfix ignora el parámetro mynetworks_style.) Puede ofrecer una o más direcciones IP y/o usar su notación de red o de máscara de red (por ejemplo 151.164.28.0/28).

Este parámetro es útil cuando desea dirigir correos a equipos que están fuera de su red, por ejemplo, para ejecutivos que trabajan desde casa, viajantes, etc.

Haremos algunos otros cambios en `/etc/postfix/main.cf` más tarde en este capítulo para añadir autentificación y encriptación de contraseñas.

Probando el correo

Con la configuración apropiada, puede recibir y enviar correo desde su cuenta. El siguiente es un ejemplo de dos mensajes de prueba enviado por uno de los autores de este libro. Primero, usamos una cuenta de Gmail para enviar un mensaje de correo a una cuenta de usuario en el sistema `server2.centralsoft.org`. Leímos el mensaje desde la línea de comandos usando el comando estándar de Linux mail:

```
~$ mail
Message 1:
Date: Tue, 11 Jul 2006 17:38:32 -0500
From: "Tom Adelstein" <tadelstein@gmail.com>
To: tadelste@server2.centralsoft.org
Subject: Prueba de SMTP
Estamos enviando este correo para probar la funcionalidad del servidor
de correo a la hora de enviar y recibir correo SMTP simple.
```

Luego contestamos al correo original y lo recibimos en la cuenta de Gmail:

```
Delivered-To: tadelstein@gmail.com
Received: from server2.centralsoft.org
   Tue, 11 Jul 2006 16:10:44 -0700 (PDT)
To:tadelstein@gmail.com
```

```
Subject: Re: Prueba de SMTP
In-Reply-To
tadelste@server2.centralsoft.org (Tom Adelstein)

Estamos enviando este correo para probar la funcionalidad del servidor
de correo a la hora de enviar y recibir correo SMTP simple.
```

Usar el comando mail es una manera primitiva de gestionar grandes volúmenes de correo, incluso desde una cuenta local. Una alternativa es mutt, que tiene una interfaz robusta y bastantes más funcionalidades. Como administrador, podría usar algunos de estos agentes de línea de comando cuando reciba correo en su cuenta de servicio.

Añadiendo autentificación y encriptación

Ahora que ya hemos configurado un servidor SMTP por defecto, ¿qué podemos hacer con Postfix? En esta sección añadiremos autentificación (usando SASL) y encriptación (usando TLS) a nuestro archivo de configuración, `/etc/postfix/main.cf`. Con autentificación, nos aseguraremos que sólo usuarios con las credenciales adecuadas puedan usar nuestro servidor SMTP. Con encriptación, nos aseguraremos que no enviamos ni ID de usuario ni contraseñas a través de la red en texto claro.

Autentificación SASL

La figura 5.1 representa un grupo de usuarios móviles que necesitan obtener el correo a través de un servidor de correo fuere de la red local del servidor. Este es un escenario común. Para distinguir a estos usuarios legítimos de los spammers, necesita un mecanismo de seguridad. La capa SASL desarrollada como parte del proyecto Cyrus de la Universidad de Carnegie Mellon, ofrece Postfix con un medio para identificar las fuentes de correo enviadas al servidor y control del correo procesado.

Nota: Los administradores del sistema pueden usar SASL para añadir autentificación en muchas de las operaciones cliente/servidor, pero cada servicio que use SASL en un sistema operativo Linux necesita un archivo de configuración diferente. No puede instalar SASL y configurarlo para todo el sistema.

¿Cómo se convirtió SASL en parte de la solución Postfix? Para encontrar una respuesta, hemos de remontarnos hasta 1999, cuando la IETF escribió un estándar llamado Extensión del servicio SMTP para autentificación. Verá este trabajo con el acrónimo de ESMTP, por ejemplo, está en la primera línea de archivo `/etc/postfix/main.cf` (véase Tabla 5.1). ESTMP evita que una gran cantidad de remitentes y/o atacantes usen MTA desconocidos como pasarelas. También ofrece seguridad autentificando a los usuarios y sus actividades. La IETF basó la extensión ESMTP en SASL. Como parte del protocolo SMTP, ESMTP simplemente añade un comando llamado AUTH a los comandos del servidor usados para intercambiar datos. El framework de autentificación SASL permite una variedad de formas para almacenar e intercambiar credenciales de usuario. Puede usar contraseñas Linux (`/etc/passwd`, `/etc/shadow` o Módulos de Autentificación independientes), archivos independientes, o servicios externos como LDAP, Kerberos o sasldb (un directorio creado por el proyecto Cyrus e incluido con SASL).

En este capítulo, mostraremos dos formas de usar Postfix con SASL. Primero, configuraremos un método simple que funciona bien en sitios pequeños donde puede dar a cada usuario de correo una cuenta de usuario en el servidor Linux, este método usa PAM, la autentificación por defecto usada en estos casos. Luego, configuraremos un sistema más complejo que le permite autentificar usuarios que no tienen cuentas en el servidor.

Nota: La autentificación puede considerarse un proceso de dos etapas. Primero, hay que asegurarse de que el usuario es quien dice ser. Luego, hay que ofrecerle el servicio pedido, que puede ser una interfaz de comandos (bash, tcsh, zsh, etc.) o una sesión X Window ejecutándose bajo su identidad.

Configurando Postfix con SASL para autentificar usuarios con cuentas

Afortunadamente, Debian incluye SASL con Postfix. Puede usar las librerías SASL de Debian para permitir a los usuarios móviles autentificarse desde fuera de la red. En el siguiente ejemplo, usaremos SASL para verificar que las personas que están intentando conectarse tienen cuentas válidas en el servidor Linux, es decir, nuestro sistema permitirá conectarse y enviar correo sólo a las personas con cuentas en el servidor. Usaremos PAM, el mecanismo por defecto para la autentificación en Linux para hacer esto.

Cuando instaló los paquetes anteriormente, incluyó las extensiones y las librerías SASL necesarias (postfix-tls, libsasl2, sasl2-bin y libsasl2-modules). Ahora necesita configurar `/etc/postfix/main.cf`. Primero, le mostraremos cómo añadir parámetros al archivo usando comandos postconf; luego, le mostraremos una alternativa consistente en editar `/etc/postfix/main.cf` directamente. Active la autentificación en el servidor SMTP Postfix añadiendo los parámetros smtpd (servidor) a su archivo de configuración `main.cf` con este comando postconf:

```
# postconf -e 'smtpd_sasl_auth_enable = yes'
```

Luego, añada un parámetro para ajustar los clientes no estándar que no procesan la autentificación SMTP correctamente:

```
# postconf -e 'broken_sasl_auth_clients = yes'
```

El parámetro `smtod_sasl_security_options` le permite controlar todos los mecanismos de autentificación cuando los diversos clientes se conectan a su servidor SMTP. El siguiente bloque de configuración bloquea completamente la autentificación anónima:

```
# postconf -e 'smtpd_sasl_security_options = noanonymous'
```

Postfix no permite el envío de correo sin autorizar por defecto. Por lo que, para permitir a sus usuarios de correo trabajar en Internet, necesita añadir otro parámetro (nota: esto debería aparecer en una sola línea):

```
# postconf -e 'smtpd_recipient_restrictions =
permit_mynetworks,permit_sasl_authenticated,reject_unauth_destination'
```

Finalmente el parámetro `smtpd_sasl_loca_domain` establece el nombre del dominio local de autentificación. Por defecto, Postfix considera el nombre de la máquina como el nombre de dominio local de autentificación. Para usar el comportamiento por defecto, especifique una cadena nula:

```
# postconf -e 'smtpd_sasl_local_domain ='
```

Esto completa la configuración SASL para Postfix. De manera alternativa, en lugar de ejecutar los diferentes comandos postconf anteriores, tiene la opción de editar el archivo `/etc/postfix/main.cf`, añadir las siguientes entradas y finalmente recargar Postfix.

```
smtpd_sasl_local_domain = $myhostname
smtpd_sasl_auth_enable = yes
broken_sasl_auth_clients = yes
smtpd_sasl_security_options = noanonymous
smtpd_recipient_restrictions =
   permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination
smtpd_sasl_local_domain =
```

Ha terminado de configurar SASL y ahora ya puede empezar a usarlo. Antes vamos a ver los últimos pasos, ejecute este comando para crear un archivo de configuración SASL en el directorio donde Postfix buscará (la –p evita un error si el directorio ya existe):

```
# mkdir -p /etc/postfix/sasl
# cd /etc/postfix/sasl
```


Cree el archivo `smtpd.conf` con estas dos líneas:

```
pwcheck_method: saslauthd
mech_list: plain login
```

Ahora puede reiniciar Postfix:

```
# postfix reload
```

El demonio saslauthd

En el archivo `smtpd.conf`, hemos especificado saslauthd como nuestro método para verificar las credenciales del usuario. ¿Por qué?

Nuestro sistema de contraseñas usa PAM, y los procesos sin privilegios no tienen acceso a los archivos de contraseña. Debido a que la cuenta de servicio de Postfix se ejecuta con privilegios limitados, no puede autentificar directamente a los usuarios.

Las librerías SASL que se distribuyen con Debian solventan esta situación añadiendo un demonio de autentificación llamado saslauthd, que maneja las peticiones por Postfix. El demonio se ejecuta con privilegios de superusuario en un proceso separado de Postfix, por lo que alguien que comprometa la seguridad del servidor de correo no podrá disfrutar de los privilegios de saslauthd.

saslauthd no se comunica fuera de nuestra red, el impacto de seguridad del demonio es mínimo incluso si se usaran contraseñas en texto claro. saslauthd necesita las contraseñas actuales porque usa el mismo servicio de autentificación que el que usted usar para abrir una sesión de consola en Linux.

Ahora, vamos a configurar saslauthd para ejecutar el servidor de correo. Las siguientes direcciones corresponden a Debian, pero puede hacer más o menos las mismas cosas con los directorios y los comandos de otros sistemas Linux.

El puerto de Postfix en Debian se ejecuta en entorno chroot en `/var/spool/postfix`. Por consiguiente, necesita poner el demonio saslauthd en el mismo espacio de nombres. Siga los siguientes pasos:

1. Cree el directorio necesario para el demonio:

```
# mkdir -p /var/spool/postfix/var/run/saslauthd
```

2. Edite `/etc/default/saslauthd` para activar saslauthd. Elimine la marca de comentarios de la línea START=yes y luego añada la línea:

```
PARAMS="-m /var/spool/postfix/var/run/saslauthd -r"
```

3. Su archivo debe quedar así:

```
# This needs to be uncommented before saslauthd will be run automatically
START=yes
```

```
PARAMS="-m /var/spool/postfix/var/run/saslauthd -r"
# You must specify the authentication mechanisms you wish to use.
# This defaults to "pam" for PAM support, but may also include
# "shadow" or "sasldb", like this:
# MECHANISMS="pam shadow"
MECHANISMS="pam"
```

4. Lo siguiente que tiene que hacer es editar `/etc/init.d/saslauthd` para cambiar la ubicación del archivo con el ID del proceso de saslauthd. Cambie el valor de PIDFILE por lo siguiente:

```
PIDFILE="/var/spool/postfix/var/run/${NAME}/saslauthd.pid"
```

5. Luego ejecute saslauthd:

```
# /etc/init.d/saslauthd start
```

Si usa una distribución Linux distinta de Debian, trabajará con comandos, directorios y archivos diferentes. Por ejemplo, en muchos sistemas la forma estándar de iniciar saslauthd por primera vez es el comando:

```
# saslauthd -a pam
```

De no hacerlo así, Debian especificará el uso de PAM a través del archivo de configuración.

Configurando Postfix con SASL para autentificar usuarios sin cuentas

Usar el archivo de contraseñas para la autentificación en un sistema Linux requiere que cada persona que recibe el correo a través del servidor tenga una cuenta de usuario. Obviamente, esta solución carece de escalabilidad y requiere mucho tiempo administrativo. Para soportar usuarios que no tienen cuentas en el servidor SMTP, SASL le permite usar otras opciones de almacenamiento; las opciones más populares son sasldb, LDAP, Kerberos y MySQL. El demonio saslauthd no se ejecuta cuando Postfix usa uno de estos métodos: no se necesita un programa distinto con privilegios de superusuario porque SASL no necesita acceder al archivo de contraseñas del sistema operativo. Cuando usa saslauthd, está limitado a la transmisión de contraseñas en texto plano y a la autentificación vía login. Por tanto, Postfix también ofrece un método auxprop alternativo, que soporta los métodos de autentificación de texto plano, login, CramMD5, DigestMD5, OPT y NTLM. De todos los mecanismos de autentificación discutidos en este capítulo, LDAP es el más robusto y escalable, pero tiene la limitación de que usa contraseñas en texto claro. Para solucionar este problema, los administradores normalmente usan la capa TLS para encriptar las contraseñas y transmitirlas desde el cliente al servidor (como se discutirá en la siguiente sección). La combinación de LDAP y TLS actualmente es la mejor opción de seguridad.

En una red de ordenadores pequeña, sasldb puede ofrecer una solución simple válida para unos cuantos usuarios. Para sitios más grandes, con más usuarios, probablemente MySQL sea la solución más escalable y más fácil de gestionar.

El método sasldb y el MySQL necesitan que instale un tipo especial de plugin llamado auxiliary property. Si configura sasldb o MySQL, tiene que editar el archivo `smtpd.conf` y cambiar la línea:

```
pwcheck_method: saslauthd
```

por la siguiente, que ofrece un framework para el citado plugin:

```
pwcheck_method: auxprop
```

Encriptación TLS

El inconveniente de usar el método auxprop para la validación de usuarios es que, sin protecciones adicionales, usa validación de texto plano. Cuando es usted el que entra en su propia estación de trabajo, no hay ningún problema, pero cuando envía un ID de usuario y una contraseña a través de una red en texto plano para acceder al correo, ya sea una red local o Internet, cualquiera puede obtener de manera fácil sus credenciales. Anteriormente ya discutimos acerca de usar TLS, una versión actualizada de la encriptación SSL, para enviar contraseñas desde su estación de trabajo a su servidor de correo de manera segura. Aquí, vamos a extender esa solución para encriptar información de identificación creando un certificado a través de Open SSL.

> **Nota:** La sección previa de SASL y la sección actual tratan de la seguridad, aunque con diferentes objetivos. Mientras que la sección de SASL maneja la autentificación, que determina quién tiene el derecho de enviar correo a través de su servidor. Esta sección trata la protección de contraseñas, que asegura que los intrusos potenciales no pueden leer las credenciales secretas de los usuarios. Necesitará ambos servicios para un correo electrónico seguro.

Empiece por crear un directorio para los certificados SSL. Cree un subdirectorio justo debajo de la ubicación de Postfix en Debian:

```
# mkdir /etc/postfix/ssl
# cd /etc/postfix/ssl/
```

Luego, genere dos certificados y dos claves de encriptación. Necesita una clave primaria que nadie conoce y una clave pública que le permite a los otros enviarle credenciales seguras. Comience con la clave del servidor:

```
# openssl genrsa -des3 -rand /etc/hosts -out smtpd.key 1024
293 semi-random bytes loaded
```

```
Generating RSA private key, 1024 bit long modulus
......................................++++++
.....................................++++++
e is 65537 (0x10001)
Enter pass phrase for smtpd.key:
Verifying - Enter pass phrase for smtpd.key:
```

Cambie los permisos del archivo resultante que contiene la clave OpenSSL del servidor:

```
# chmod 600 smtpd.key
```

Luego, genere otra clave y otro certificado

```
# openssl req -new -key smtpd.key -out smtpd.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]: centralsoft.org
Organizational Unit Name (eg, section) []: web
Common Name (eg, YOUR name) []:
Email Address []:
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []: cso
```

Nota: Existe un debate para dirimir si los certificados que se autogeneran deberían pedir información o no. Nosotros recomendamos que introduzca la información apropiada para su sitio de producción.

Los siguientes comandos generan un clave de firma y cambian las claves existentes por las nuevas:

```
# openssl x509 -req -days 3650 -in smtpd.csr -signkey smtpd.key -out \
smtpd.crt
Signature ok
subject=/C=US/ST=Texas/L=Dallas/O=centralsoft.org/OU=web/CN=Tom_Adelstein/
emailAddress=tom.adelstein@gmail.com
Getting Private key
Enter pass phrase for smtpd.key:
# openssl rsa -in smtpd.key -out smtpd.key.unencrypted
Enter pass phrase for smtpd.key:
```



```
writing RSA key
# mv -f smtpd.key.unencrypted smtpd.key
# chmod 600 smtpd.key
# openssl req -new -x509 -extensions v3_ca -keyout cakey.pem -out \
cacert.pem -days 3650
Generating a 1024 bit RSA private key
....................++++++
.........................++++++
writing new private key to 'cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:
Email Address []:
```

Ahora necesita indicarle a Postfix las claves y los certificados de los siguientes comandos postconf:

```
# postconf -e 'smtpd_tls_auth_only = no'
# postconf -e 'smtp_use_tls = yes'
# postconf -e 'smtpd_use_tls = yes'
# postconf -e 'smtp_tls_note_starttls_offer = yes'
# postconf -e 'smtpd_tls_key_file = /etc/postfix/ssl/smtpd.key'
# postconf -e 'smtpd_tls_cert_file = /etc/postfix/ssl/smtpd.crt'
# postconf -e 'smtpd_tls_CAfile = /etc/postfix/ssl/cacert.pem'
# postconf -e 'smtpd_tls_loglevel = 1'
# postconf -e 'smtpd_tls_received_header = yes'
# postconf -e 'smtpd_tls_session_cache_timeout = 3600s'
# postconf -e 'tls_random_source = dev:/dev/urandom'
```

El archivo `/etc/postfix/main.cf` debería quedar así:

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete version
smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
# appending .domain is the MUA's job.
append_dot_mydomain = no
# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h
myhostname = server1.example.com
```

```
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = server1.example.com, localhost.example.com, localhost
relayhost =
mynetworks = 127.0.0.0/8
mailbox_command = procmail -a "$EXTENSION"
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
smtpd_sasl_local_domain =
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_recipient_restrictions = permit_sasl_authenticated,
permit_mynetworks,reject_
unauth_destination
smtpd_tls_auth_only = no
smtp_use_tls = yes
smtpd_use_tls = yes
smtp_tls_note_starttls_offer = yes
smtpd_tls_key_file = /etc/postfix/ssl/smtpd.key
smtpd_tls_cert_file = /etc/postfix/ssl/smtpd.crt
smtpd_tls_CAfile = /etc/postfix/ssl/cacert.pem
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
```

Ahora ya puede reiniciar el demonio Postfix:

```
# /etc/init.d/postfix restart
Stopping mail transport agent: Postfix.
Starting mail transport agent: Postfix.
```

Configurando los agentes de entrega de correo POP3 e IMAP

En esta sección añadiremos agentes de entrega de correo para complementar a Postfix. Use el siguiente comando en Debian para añadir un servidor IMAP y uno POP3.

```
# apt-get install ipopd-ssl uw-imapd-ssl
```

Hemos escogido ipopd-ssl para ofrecer agentes de entrega de correo POP2 y POP3 y uwimapd-ssl para IMAP. No se preocupe por el sufijo ssl, tanto los paquetes que ofrecen servicios sin encriptar como los que los ofrece encriptados. El IMAP estándar usa el puerto 143 y POP3 usa el puerto 110. Los protocolos encriptados y los puertos son POP3S (puerto 995) e IMAPS (puerto 993).

Originalmente creado en la Universidad de Washington, el paquete ipopd-ssl está mantenido por Debian. Sólo necesita instalarlo, básicamente configura el uso del directorio de correo que existe en el servidor de correo al igual que el que configuramos en el capítulo anterior. Los ISP aún continúan usando POP3, pero las empresas no suelen usarlo.

uw-imapd-ssl ofrece un servidor IMAP. Aunque requiere más espacio en disco, IMAP es superior a POP porque deja el correo en el servidor y permite a los usuarios ver mensajes desde cualquier ubicación con acceso a Internet y un cliente de correo. No conocemos ningún cliente de correo actual que no pueda procesar IMAP, por lo que la mayoría de los usuarios pueden usarlo.

También puede ofrecer correo Web en su servidor de correo usando SSL (https), permitiendo así que los usuarios accedan a su correo desde un navegador Web.

Nota: En nuestra configuración, los usuarios necesitan cuentas de Linux estándar en su servidor de correo, incluso aunque lean su correo desde un cliente en otro sistema. Postfix normalmente permite entrega local en un dominio, pero requiere una pasarela (como se discutió en la sección previa) si los usuarios están fuera del dominio.

uw-imapd tiene ventajas y desventajas. Por una parte, usa el estilo Unix para el buzón de correo, lo que mantiene el correo de todos los usuarios en un archivo único en su directorio de inicio. Este servicio es fácil de administrar.

Por la otra parte, uw-imapd no permite a los usuarios virtuales o a aquellos que no tenga cuenta y directorio de inicio acceder al correo. Además, a muchos administradores no les gusta el formato de almacenamiento de mbox, y prefieren el formato jerárquico de maildir. Al ser un formato de archivo único mbox permite acceder a la vez a una sola aplicación, lo que implica bloqueos y hace que el sistema vaya más lento.

Nota: El bloqueo de archivos es un mecanismo que obliga a que sólo un usuario o proceso pueda acceder al archivo en un momento dado. El propósito del bloqueo es evitar actualizaciones conflictivas.

Muchas personas consideran que el bloqueo de archivos es un problema en el caso del correo. Muchos sistemas de archivos distribuidos carecen de mecanismos de bloqueo. Algunas personas creen que incluso el bloqueo de archivos es insuficiente para evitar fallos ocasionales en mbox. Con Linux, el fallo es posible si un proceso de correo falla en medio de la actualización de un buzón.

El formato maildir, sin embargo, permite acceso concurrente a múltiples aplicaciones y no necesita bloqueo de archivos. Otros servidores IMAP, como Cyrus, Courier y Dovecot, usan el formato maildir para permitir a los usuarios virtuales

y a los usuarios sin cuenta ni directorio de inicio acceder al correo. Configurado junto con Postfix, las cuentas de usuario sólo tienen buzones de correo. Esto permite a los administradores mantener el MTA y el MDA si tener que gestionar cuentas de usuario estándar en el servidor.

Otros servidores IMAP distintos de uw-imapd son difíciles de configurar y necesitan una buena dosis de experiencia, por lo que decida por usted mismo si el tamaño de su organización justifica su uso. De ser así, necesitará buscar otros recursos para informarse, como el libro "The Book of Postfix" por Ralf Hildebrandt y Patrick Katter.

Configuración del cliente de correo

En nuestra introducción previa al archivo de configuración de Postfix `/etc/postfix/main.cf`, dejamos en manos del cliente de correo del usuario la decisión de añadir el nombre de dominio cuando un usuario escribía un correo desde su cuenta de correo:

```
append_dot_mydomain = no
```

Este es el comportamiento de la mayoría de los clientes, que pueden añadir un dominio como `@centralsoft.org` cuando el usuario introduce un nombre de cuenta en el campo Para de un mensaje de correo electrónico.

Si configura Postfix para usar encriptación, como se mostró anteriormente en este capítulo, el usuario de corro también tiene que configurar su MUA para usar encriptación TLS a la hora de enviar correo. Los clientes más modernos soportan esto y ofrecen una interfaz gráfica para activar TLS y usarlo con el servidor de correo saliente.

Cuando no está en una red definida por Postfix y es cliente estático (al contrario que el cliente móvil), use el servidor SMTP de correo saliente de su ISP. En este caso, debería seleccionar TLS si su servidor ofrece su uso. En la gran mayoría de los casos, su IS y su contraseña viajarán en texto claro a través de las líneas de su ISP. Para el servicio de recepción de correo, necesitará configurar un servidor entrante con DNS, como se vio anteriormente. Como breve recordatorio, deberá usar registros MX para hacer esto. Un registro MX normal sería:

```
MX 10 server1.centralsoft.org.
```

Este registro indica claramente que el correo enviado al dominio `centralsoft.org` debería ser entregado al `server1.centralsoft.org` (que es el servidor de correo del dominio).

Qué es lo próximo

En este punto, ha instalado y configurado Postfix y un servicio IMAP y otro POP3. Ya tiene los componentes esenciales de un sistema de correo que puede usarse en un entorno corporativo.

Si esta es su primera toma de contacto con el correo, ahora podrá comprender por qué las empresas gastan grandes cantidades de dinero en comprar licencias de sistemas. También podrá comprender por qué pagan a una docena o más de administradores de sistemas para que gestionen las infraestructuras de comunicación por correo electrónico. Esta área requiere una experiencia especial. Después de que haya asimilado la información de este libro, podrá estudiar los demás componentes de los sistemas de correo avanzados. Debería saber cómo instalar y configurar un servidor de correo escalable y seguro y cuánto esfuerzo es necesario para adquirir experiencia en este campo. También necesitará aprender a gestionar servicios de directorio, como OpenLDAP o Fedora Directory Server, para validar una gran cantidad de usuarios y ofrecer un listado de los usuarios de correo de su empresa.

El próximo capítulo revisa el servicio que mucha gente considera como el más crítico de una organización: un servidor Web. Después, introduciremos la configuración de uno de los servidores Web más populares, Apache, y procederemos a instalar un amplio rango de funcionalidades importantes, como soporte para sitios Web dinámicos y recolección de estadísticas, además le proporcionaremos algunos consejos para resolver problemas.

Capítulo 6

Administrando Apache



En este capítulo, levantaremos un servidor Web Linux desde cero. Aprenderá a:

- Instalar y configurar Apache, PHP y MySQL.
- Gestionar múltiples sitios Web con hosts virtuales.
- Encriptar páginas sensibles con SSL.
- Activar tecnologías de servidor y scripts CGI.
- Hacer pruebas de rendimiento y de seguridad.
- Instalar vlogger y Webalizer para ver las estadísticas del sitio.
- Instalar Drupal, un sistema gestor de contenidos que será muy útil en muchos entornos y que usa muchos de estos elementos.

Este capítulo describe un entorno con un servidor Web, en el capítulo siguiente, veremos cómo configurar un par de servidores Web para balancear el trabajo.

Los servidores Web son grandes y complejos, y al configurarlos no está claro cómo o por qué se hacen determinadas cosas. A lo largo del camino, le mostraremos por qué hemos elegido algunas alternativas en detrimento de otras. Para que las explicaciones sean breves y claras, usaremos los procedimientos estándar de Debian. Dotaremos de seguridad al entorno a medida que avancemos, para resaltar que la seguridad es un aspecto que debe tenerse en cuenta desde el principio. Al final del capítulo, podrá encontrar una sección para la solución de errores comunes.

Archivos estáticos y dinámicos

Un sitio Web básico está compuesto por archivos: HTML, gráficos, JavaScript, hojas de estilos y otros tipos. El contenido de estos archivos es estático, no cambian en el servidor, y el único trabajo del servidor Web es devolverlos cuando el

navegador lo solicite. Un servidor Web necesita sólo una pequeña configuración para servir archivos estáticos.

Muchos sitios también tienen contenido dinámico, incluso generación automática de contenido, control de acceso y almacenamiento en bases de datos. La manera más fácil para hacer archivos HTML dinámicos es usar la tecnología de servidor (SSI), que equivale a añadir comentarios formateados mediante HTML que Apache interpreta para mostrar su valor o incluir contenidos de otros archivos HTML. La inclusión de archivos SSI es una forma fácil para definir una cabecera y un pie de página común a todo el sitio, por ejemplo.

SSI tiene límites, aunque una gran mayoría de sitios dinámicos usan programas *Common Gateway Interface* (CGI). Estos programas ejecutables pueden escribirse en cualquier lenguaje que Linux soporte, aunque las opciones más populares son los lenguajes de script como Perl, PHP, Python y Ruby, seguidos de Java. CGI es un protocolo que especifica cómo deben intercambiar las peticiones y las respuestas el cliente y el servidor. Cuando los primeros CGI aparecieron en la Web, eran totalmente independientes de los servidores Web. Cada petición provocaba que el servidor Web iniciara un nuevo proceso CGI. El coste de arranque incrementaba el tiempo de respuesta del sistema, por lo que se desarrollaron alternativas. La gente a menudo confunde el protocolo CGI con este método de implementación anticuado y por ello piensan que CGI es lento. Sin embargo, el estándar CGI no define la implementación. Existen métodos más rápidos que siguen el mismo protocolo CGI.

Un método más rápido es FastCGI, que inicia el programa CGI como un proceso aparte y gestiona dos vías de comunicación entre dicho proceso y el servidor Web. Evitando así el coste de recarga, además la separación del proceso asegura que en el caso de que el CGI falle, no llevará asociada la caída del servidor. No obstante, FastCGI tiene un inconveniente: al igual que otros programas CGI, no puede acceder a los entresijos del servidor Web, algo que es necesario en aplicaciones complejas.

Algunos programas CGI han evolucionado en forma de módulos Apache que se cargan como parte del propio servidor Web: el intérprete de Perl se convirtió en `mod_perl` o PHP se ha convirtió en `mod_php`. El rendimiento de los programas FastCGI y de los módulos Apache es similar, y los módulos tienen tanto ventajas como desventajas. Tienen acceso a todas las estructuras de datos y funciones internas del servidor, por lo que pueden usarse en varias etapas de la transacción Web, pero no para generar contenido HTML. Sin embargo, los módulos incrementan el tamaño y el uso de la memoria del servidor Web, y un fallo en el módulo puede provocar la caída del servidor Web.

Instalación básica de LAMP

La instalación estándar de LAMP (Linux, Apache, MySQL, PHP/Perl/Python) usa módulos Apache para ejecutar las funciones CGI. Esta aproximación funciona muy bien y es fácilmente escalable, aunque todo tiene sus límites. Apuntaremos algunos de esos límites en este capítulo, pero puede saltarse estas secciones si prefiere aprender de la experiencia. Ya tenemos la L, ahora exploraremos las A; la M y la P vendrán un poco después. Apache no es el servidor Web más rápido, ni el más fácil de configurar, ni el más seguro, pero es muy superior al resto de servidores. Según Netcraft, más del 60 por 100 de los sitios Web públicos usan Apache (`http://news.netcraft.com/archives/web_server_survey.html`). Apache se ejecuta en Linux, Mac OS X y otros sistemas basados en UNIX, además de casi todas las versiones de Microsoft Windows.

Al igual que otros programas Unix, Apache puede levantarse con todos sus módulos, obteniendo así un programa muy grande (con ligadura estática) o puede trabajar con módulos que se cargan en memoria a medida que se necesitan (objetos dinámicos compartidos o DSO). El método DSO es más fácil y más flexible, puesto que le permite añadir módulos una vez que Apache ha sido levantado. La instalación de Debian para PHP y para otros módulos usa el método DSO.

Instalación

En esta sección, instalaremos Apache, PHP y MySQL. Usaremos la instalación por defecto para cada uno de ellos para asegurarnos que se ejecutan correctamente. En la sección siguiente, exploraremos los archivos de configuración de Apache y cómo personalizar nuestra instalación.

Apache

Necesita ser usuario root para instalar paquetes. Primero consiga el servidor Apache:

```
# apt-get install apache2
```

Esto debería instalar Apache e iniciarlo. ¿Funcionó? Para saberlo, introduzca su URL en un navegador Web. Para los ejemplos de este capítulo, usaremos el nombre de nuestro servidor de pruebas (`http://server1.centralsoft.org`). Cuando vea esta URL en los ejemplos, debe sustituirla por la URL de su propio servidor. Si está ejecutando el navegador en la misma máquina en que está alojado el servidor Web y tiene problemas con la resolución DNS de su servidor de nombres, puede usar `http://localhost` o `http://127.0.0.1`. Si lo está probando desde fuera, puede usar la dirección IP del servidor, en este caso `http://70.253.158.41`.

Introduzca la URL del servidor en un navegador Web y podrá ver una página que empieza así:

```
Si puede ver esto, es que la instalación del servidor Web de Apache se
ha instalado de manera satisfactoria en su sistema. Ahora tiene que
añadir contenido a este directorio y reemplazar esta página.
```

El navegador debería mostrar que Apache ha redirigido la página que introdujo por otra: `http://server1.centralsoft.org/apache2-default`.

Explicaremos esto un poco después cuando indaguemos en los archivos de configuración de Apache pero, por ahora, nos conformaremos con crear nuestro primer archivo Web. Vaya al directorio que Apache considera el directorio raíz de su sitio y cree un archivo simple de texto:

```
# cd /var/www
# echo probando > test.html
```

Luego introduzca la URL (por ejemplo, `http://server1.centralsoft.org/test.html`) en su navegador. Podrá ver la palabra probando en la pantalla. Su servidor Apache se está ejecutando sin restricciones de acceso, es decir, sirviendo cualquier archivo y directorio que exista en `/var/www`.

PHP

PHP es el módulo CGI de Apache más popular. En este capítulo, usaremos PHP 4, que todavía es más popular que su sucesor, PHP 5. Usar PHP es una buena forma de crear páginas dinámicas, y la gran librería de módulos PHP ofrece muchas funciones útiles. Comience obteniendo el programa PHP y las librerías:

```
# apt-get install php4
```

Ahora consiga el módulo PHP para Apache, `mod_php`. Este comando instalará `mod_php` y le indicará a Apache que le permite ejecutar archivos con sufijo .php:

```
# apt-get install libapache2-mod-php4
```

Cree este script PHP de prueba y guárdelo en `/var/www/info.php`:

```
<?php
phpinfo( );
?>
```

Luego introduzca la URL del script (`http://server1.centralsoft.org/info.php`) en su navegador. Debería ver una página con tablas de información sobre la configuración de PHP. Esta información ofrece mucha información acerca de su equipo y que no tiene que compartir con el resto del mundo, por lo que puede borrarlo después de probarlo. Si no es capaz de ver nada, échele un vistazo al final de este capítulo.

Entre tanto, si usted es novato, acaba de escribir su primer script CGI. (En la sección de CGO, ofreceremos más detalles acerca de cómo los servidores Web ejecutan programas externos y scripts.)

MySQL

Si no necesita una base de datos, tendrá una plataforma LAP y podrá saltarse esta sección. Para obtener la plataforma LAMP, obtenga el servidor de base de datos MySQL y el módulo PHP para MySQL:

```
# apt-get install mysql-server
# apt-get install php4-mysql
```

Esto es todo lo que necesita para crear scripts PHP que accedan a un servidor de bases de datos MySQL, pero también instalaremos el cliente estándar de línea de comandos para MySQL (mysql) para que nos ayude a probar la base de datos sin tener que usar PHP o Apache.

```
# apt-get install mysql-client
```

Nota: Si ejecuta el cliente mysql pero no especifica un nombre de cuenta de MySQL con la opción -u, intentará usar el nombre de su cuenta Linux. En nuestros ejemplos, hemos accedido como root, por lo que el nombre sería root. Sucede que la cuenta de administrador de MySQL también se llama root, y tiene el control completo de la base de datos. Sin embargo, las cuentas root de MySQL y de Linux no tienen nada que ver la una con la otra. MySQL almacena los nombres de cuenta y las contraseñas en la propia base de datos.

Use este comando para ver si el servidor de la base de datos se está ejecutando:

```
# mysql -u root
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 5 to server version: 4.0.24_Debian-10sarge2-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+----------+
| Database |
+----------+
| mysql    |
| test     |
+----------+
2 rows in set (0.00 sec)

mysql> quit;
```

configuración de Apache por el sistema de archivos, lo que hace difícil saber qué opciones tienen efecto sobre un directorio en un momento dado. Si no necesita archivos .htaccess, no los use. Están desactivados por defecto.

Directivas de archivos de configuración

Cada archivo de configuración de Apache está divido en secciones que contienen directivas Apache (comandos u opciones) y sus valores. Algunas directivas son parte del núcleo de Apache, mientras que las otras sólo las usan módulos específicos. Si una directiva se refiere a un módulo que no ha configurado, Apache fallará al iniciarse y se escribirá un mensaje que indique las líneas incorrectas en el archivo log de errores.

Cuando Apache se esté ejecutando correctamente, podrá ver las directivas que se están usando si introduce este comando:

```
# /usr/sbin/apache2 -L
```

Al final de este capítulo encontrará la ayuda para diagnosticar problemas en el servidor Web.

Suponiendo que el archivo de prueba funcionó, ahora puede pasar a configurar Apache. Lo siguiente es el contenido por defecto del archivo de configuración `/etc/apache2/sites-enabled/000-default`. Las secciones comienzan y terminan con etiquetas al estilo HTML, como se puede observar aquí:

```
<VirtualHost *>
...
</VirtualHost>
```

Aquí hay una copia del archivo comentado:

```
# Answer to any name or IP address:
NameVirtualHost *

# For any virtual host at any address, any port:
<VirtualHost *>
   # If Apache has problems, whom should it contact?
   ServerAdmin webmaster@localhost

   # Our web site files will be under this directory:
   DocumentRoot /var/www/

   # Overall directives, in case we move DocumentRoot
   # or forget to specify something later:
   <Directory />
      # Lets Apache follow symbolic links:
      Options FollowSymLinks
      # Disables .htaccess files in subdirectories:
      AllowOverride None
   </Directory>

   # DocumentRoot itself:
   <Directory /var/www/>
      Options Indexes FollowSymLinks MultiViews
      # Forbids .htaccess files:
      AllowOverride None
      Order allow,deny
      allow from all
      # Maps / to /apache2-default, the initial welcome
      # page that says "If you can see this...":
      RedirectMatch ^/$ /apache2-default/
   </Directory>

   # Permits CGI scripts:
   ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
   <Directory "/usr/lib/cgi-bin">
      AllowOverride None
      Options ExecCGI -MultiViews +SymLinksIfOwnerMatch
      Order allow,deny
      Allow from all
   </Directory>

   # Error log for a single site:
   ErrorLog /var/log/apache2/error.log

   # Possible values include: debug, info, notice,
   # warn, error, crit, alert, and emerg:
   LogLevel warn

   # Access log for a single site:
   CustomLog /var/log/apache2/access.log combined

   # Sends Apache and PHP version information to browsers;
   # Set to Off if you're paranoid, or have reason to be:
   ServerSignature On

   # Shows Apache docs (only to local users)
   # if you installed apache2-docs;
   # to suppress showing the documents,
   # you can comment these lines or delete them:
   Alias /doc/ "/usr/share/doc/"
   <Directory "/usr/share/doc/">
      Options Indexes MultiViews FollowSymLinks
      AllowOverride None
      Order deny,allow
      Deny from all
      Allow from 127.0.0.0/255.0.0.0 ::1/128
   </Directory>
</VirtualHost>
```

La mayor parte de los cambios que haremos en los archivos de configuración de Apache en esta sección se harán sobre este archivo. El archivo de configuración /etc/apache2/apache2.conf contiene muchas opciones de la parte servidora que no necesitan cambiarse, con algunas excepciones notables que serán las que veamos aquí.

Directivas de usuarios y grupos

Estas importantes opciones le indican a Apache que se ejecute para un ID de usuario y un ID de grupo particular. El archivo /etc/apache2/apache2.conf por defecto contiene:

```
User www-data
Group www-data
```

Cualquier archivo y directorio servido por Apache necesita ser leído por este usuario y este grupo. Los permisos de archivo y de directorio incorrectos son una de las causas más comunes de errores en Apache, como la incapacidad de ver una página (o la posibilidad de ver algo que no debería verse).

Directiva Listen

Apache normalmente responde a las peticiones en Puerto TCP 80, pero puede escuchar en otros puertos, además del 80. Es muy común usar otro puerto para pruebas, muchas personas usan el 81 porque es fácil de recordar y porque no se usa para otra cosa, para especificar uno o más puertos, use una o más directivas Listen:

```
Listen 81
```

Si va a usar encriptación SSL para algunas páginas, necesitará incluir esta directiva para usa el puerto seguro estándar:

```
Listen 443
```

Directiva DocumentRoot

Cada sitio Web tiene su raíz, que es el directorio que tienen los archivos de contenido y los scripts. Está especificado por la directiva DocumentRoot. En la configuración por defecto de Debian, está especificado en /etc/apache2/sites-enabled/000-default:

```
DocumentRoot /var/www/
```

Autentificación y autorización

Algunas partes de su sitio Web estarán abiertas a todos, pero quizás quiera restringir el acceso a ciertos visitantes. La autentificación determina quién es un

visitante. La autorización determina lo que el visitante puede hacer, como por ejemplo:

- Leer un archivo.
- Usar tecnología de servidor.
- Ejecutar un programa CGI.
- Generar un índice para un directorio que carezca de él.

En Apache, el lugar habitual para almacenar la información de autentificación es un archivo de usuario en texto plano (a menudo llamado archivo .htpasswd, después de que el programa lo modifique). El archivo de usuario contiene los ID de usuario y las contraseñas cifradas. El archivo de grupo opcional contiene los ID de grupo y los ID de usuario en texto plano. Es útil para sitios grandes debido a que permite especificar permisos para un grupo entero, en lugar de especificarlos para cada uno de los usuarios individuales.

Archivos de usuario

Como ejemplo, cree un directorio protegido por contraseña y coloque un pequeño archivo de texto dentro:

```
# cd /var/www
# mkdir secret
# cd secret
# echo "puedes verme" > file.html
```

Puesto que no lo ha protegido todavía, el archivo debería ser visible desde el navegador (http://server1.centralsoft.org/secret/file.html):

```
puedes verme
```

Ahora haga un archivo de usuarios:

```
# cd /tmp
# htpasswd -c /tmp/users jack
New password: black_pearl
Re-type new password: black_pearl
Adding password for user jack
```

Su contraseña no se mostrará cuando la introduzca. Necesitará incluir el parámetro –c la primera vez que ejecute el programa htpasswd en el archivo, para crear el archivo.

Nota: No use el parámetro –c cuando añada más usuarios, porque esto provocará que el archivo se sobrescriba.

Si quiere cambiar la contraseña de jack después, introduzca:

```
# htpasswd /tmp/users jack
New password: kraken
Re-type new password: kraken
Updating password for user jack
```

El archivo de usuario consiste en una serie de líneas que contienen un nombre de usuario y una contraseña encriptada, separados por dos puntos (:), como se muestra aquí:

```
jack:OSRBcYQ0d/qsI
```

Ahora edite el archivo de configuración del sitio Apache `/etc/apache2/sites-enabled/000-default` y añada (antes del final de la línea `</VirtualHost>`):

```
<Location /secret>
   AuthName "test"
   AuthType Basic
   AuthUserFile /tmp/users
   Order deny,allow
   require valid-user
</Location>
```

`AuthName` es obligatorio y debe ir seguido por una cadena de texto. Nosotros hemos usado "test", puede usar "" si quiere, pero por alguna razón no puede omitir la directiva. `AuthType Basic` quiere decir que estamos usando un archivo de usuario de tipo htpasswd. `AuthUserFile` especifica la ubicación de dicho archivo. La directiva `Order` indica que Apache debería denegar el acceso por defecto, y permitir el acceso sólo si el usuario está especificado en el archivo de usuarios. Por último, la directiva `require` indica que cualquier usuario del archivo de usuarios tiene permitido el acceso. Para permitir que sólo el usuario jack pueda ver el archivo que hemos creado, debería poner:

```
require jack
```

Y si quiere permitir más de un usuario, debería poner algo parecido a esto:

```
require jack will Elizabeth
```

Apache debe volver a leer su archivo de configuración para que los cambios tengan efecto:

```
# /etc/init.d/apache2 reload
```

Ahora intente acceder al archivo (`http://www.example.com/secret/file.html`) desde alguna de las cuentas listadas en el archivo de usuario. Podrá ver un cuadro de diálogo donde pone algo parecido a esto:


```
Enter username and password for "test" at server1.centralsoft.org
Username:
Password:
```

Introduzca el nombre y la contraseña (verá los asteriscos cuando introduzca la contraseña) y haga clic en **OK**. Debería ver:

```
puedes verme
```

Archivos de grupo

Otra forma de manejar muchos usuarios es usar el archivo de grupo. Cree un archivo `/tmp/groups` que contenga un nombre de grupo, dos puntos y uno o más nombres de usuario separados por espacios:

```
pirates: jack will elizabeth
```

También se puede poner el grupo y cada usuario individualmente:

```
pirates: jack
pirates: will
pirates: elizabeth
```

Luego añada una directive AuthGroupFile a 000-default:

```
<Location /secret>
   AuthName "test"
   AuthType Basic
   AuthUserFile /tmp/users
   Order deny,allow
   AuthGroupFile /tmp/groups
   require group pirates
</Locat
```

Reinicie Apache como de costumbre para que los cambios surtan efecto:

```
# /etc/init.d/apache2 reload
```

Contenedores y alias

Apache aplica restricciones de autorización a los contenedores, o a archivos y directorios del servidor. Un ejemplo de contenedor se discutió en secciones anteriores. Ahora revisaremos varias directivas de contenedores.

Rutas absolutas: Directorio

Esta directiva especifica un directorio del servidor. Aquí mostramos un ejemplo de los contenidos originales del archivo de configuración de Apache:

```
<Directory />
   Options FollowSymLinks
   AllowOverride None
</Directory>
```

Rutas relativas: Ubicación

Esta directiva especifica los archivos y los directorios relativos al directorio raíz. Por ejemplo:

```
<Location /cgi>
   Options ExecCGI
</Location>
```

permite programas CGI situados en /var/www/cgi. Veremos esto de nuevo en la sección CGI.

Reconocimiento de patrones: Archivos y Comparación de archivos

Puede que necesite un archivo o un directorio específico basándose en algún patrón de texto. Aquí mostramos un ejemplo de cómo evitar que las personas descarguen imágenes de su sitio sin autorización, para ello comprobamos dónde se originó la petición. Se usa la directiva FilesMatch, que permite especificar expresiones regulares (patrones) sin marcas de comillas:

```
# Some notes on the regular expression:
# \. means a literal dot character.
# (gif|jpg|jpeg|png) means any of these four strings.
# $ means the end of the filename.
# The regular expression will match files with the suffix
# .gif, .jpg, .jpeg, or .png.
<FilesMatch "\.(gif|jpg|jpeg|png)$">
   # Set the environment variable local to 1
   # if the referring page (the URL this image
   # was called from) is on this site.
   # Set local to 0 if the URL was on another site
   # that wants to steal our lovely images.
   SetEnvIfNoCase Referer "^http://server1.centralsoft.org/" local=1
   Order Allow, Deny
   # This checks the local variable and
   # allows access only if the referrer was local.
   Allow from env=local
</FilesMatch>
```

Alias

La directiva Alias asigna un nombre a un directorio:

```
Alias /test /tmp/test
```


En la directiva, primero va el alias (nombre alternativo), seguido de la ubicación actual del directorio. El directorio puede estar fuera de la raíz. En este caso, el archivo tmp/test/button.gif estaría accesible desde la URL http://www.example. com/test/button.gif, incluso aunque no estuviera en /var/www/test.

Límites

En un servidor ocupado, Apache puede crear muchos procesos hijos simultáneos y usar mucha memoria. Esto puede incrementar la carga media y hacer que el sistema se vuelva más lento, incluso que no responda. La tabla 6.2 muestra cómo puede limitar algunos valores de ejecución de Apache en el archivo de configuración del sitio.

Tabla 6.2. Directivas de recursos Apache.

Directiva	Valor por defecto	Uso
MaxClients	256	Máximo de peticiones simultáneas. Si llegan más peticiones, se rechazan.
MaxRequestsPerChild	0 (infinito)	Máximo de peticiones servidas antes de que un proceso hijo se reinicie. Para evitar pérdidas de memoria.
KeepAlive	on	Reaprovecha las conexiones TCP entre el cliente Web y Apache. Incrementa la velocidad de descarga de la página, puesto que devuelve todos los contenidos de la página a través de la misma conexión.
KeepAliveTimeout	15	Segundos que se deben esperar antes de que otra petición use la misma conexión.

Tecnología de servidor

La SSI puede usarse para incluir archivos de contenido, salida de programas, o contenidos de variables de entorno como parte de un archivo HTML. La sintaxis para especificar SSI en los archivos de configuración de Apache puede ser confusa. Por ejemplo, para permitir tecnología de servidor en /var/www/ssi, pero no otras opciones, cree un directorio:

```
# mkdir /var/www/ssi
```

e indique a Apache que permita esta tecnología:

```
<Location /ssi>
   Options Includes
</Location>
```

Para añadir SSI a las opciones existentes, use:

```
<Location /ssi>
   Options +Includes
</Location>
```

SSI le permite incluir archivos de contenido, pero también puede ejecutar cualquier programa e incluir su salida. Esto puede no ser muy seguro, por lo que para restringir la inclusión de archivos de contenido solamente, use:

```
<Location /ssi>
   Options IncludesNoExec
</Location>
```

Si quisiera tener archivos SSI en varios lugares en lugar de tenerlos todos confinados en este directorio, puede indicarle a Apache que asocie a tipo de archivo el sufijo SSI:

```
AddHandler server-parsed .shtml
```

Para que SSI funcione, el módulo de Apache correspondiente tiene que estar cargado, puesto que no se carga con la instalación por defecto de Apache o de PHP, tendremos que teclear lo siguiente:

```
# a2enmod include
Una vez instalado el módulo include; ejecute /etc/init.d/apache2
force-reload para activarlo.
# /etc/init.d/apache2 force-reload
```

Los comandos SSI parecen comentarios HTML. Tienen la forma:

```
<!--#command argument="value"-->
```

Los posibles valores del comando son include (incluye archivos), echo (muestra variables de entorno), exec (incluye la salida de un comando) y config (da formato a las variables de echo).

Primero vamos a probar la inclusión de archivos. Cree dos archivos:

```
# cd /var/www/ssi
# echo "top stuff" > top.html
# echo "bottom stuff" > bottom.html
```


Ahora cree el archivo `middle.shtml` con este contenido:

```
<!--#include virtual="top.html"-->
En el medio
<!--#include virtual="bottom.html"-->
```

Fíjese en que el archivo que está haciendo la inclusión (`middle.shtml`) necesita la extensión .shtml, pero los archivos que se están incluyendo (`top.html` y `bottom.html`) no. Ahora ponga en la barra de direcciones del navegador `http://server1. centralsoft.org/middle.shtml` y podrá ver:

```
arriba
en el medio
abajo
```

Si la opción Includes está definida como contenedor, SSI también puede ejecutar comandos, pero el usuario (normalmente un navegador Web) no puede pasarle ninguna directiva. La ejecución de comandos SSI se suele usar para hacer fáciles algunas cosas como el listado de directorios:

```
<!--#exec cmd="ls -l /tmp"-->
```

Otro funcionalidad de SSI es la de mostrar variables de entorno CGI y otro tipo de variables. Una forma rápida de imprimir todas estas variables es:

```
<!--#printenv-->
```

Para una variable determinada, sería:

```
<!--#echo var="DATE_GMT"-->
```

lo que mostraría:

```
Tuesday, 01-Aug-2006 02:42:24 GMT
```

Si sólo tiene archivos estáticos, o una mezcla de archivos estáticos y scripts CGI, es más seguro desactivar la ejecución de comandos SSI:

```
<Location />
   Options IncludesNoExec
</Location>
```

CGI

CGI es una forma mucho más flexible (y peligrosa) de ejecutar programas en servidores Web, puesto que los usuarios pueden pasar información a los programas. Apache tiene dos formas de especificar qué programas pueden ejecutarse como programas CGI.

Location

Las dos siguientes directivas sirven para asociar los programas CGI del directorio /var/cgi con la URL que comienza por http://server1.centralsoft.org/cgi/:

```
ScriptAlias /cgi /var/cgi
```

o

```
<Location /cgi>
   Options ExecCGI
</Location>
```

Sufijo del archivo

El método del sufijo asocia un tipo MIME (nombre estándar para tipos de archivos) con un sufijo. El módulo php usa este método para conseguir que Apache pase los archivos .pho al intérprete de mod_php:

```
AddType application/x-httpd-php .php
```

Aquí le mostramos los contenidos del archivo de configuración de Apache para mod_php (/etc/apache2/mods-enabled/php4.conf), que trata los archivos con el sufijo .phtml o .php como scripts PHP:

```
<IfModule mod_php4.c>
   AddType application/x-httpd-php .php .phtml .php3
   AddType application/x-httpd-php-source .phps
</IfModule>
```

La primera línea AddType indica que los archivos que terminen en .php, .php3 o .phtml se ejecuten como programas CGI. La segunda línea AddType le indica a Apache que imprima el contenido de los archivos con sufijo .phps en lugar de ejecutarlos y devolver su salida. Los autores de sitios Web pueden usar esto para ejecutar un script (.php) y permitir que los usuarios vean el código fuente (.phps). Si de manera accidental usa el sufijo .phps en lugar de .php, su script no se ejecutará, sino que su contenido se mostrará por pantalla.

Nunca ponga un intérprete de Perl, PHP o una shell de Linux en un directorio CGI. Cualquiera podría ejecutarlos con permisos de usuario y de grupo de Apache.

Cuando probamos la instalación PHP anteriormente, creamos un pequeño programa PHP:

```
<?php
phpinfo( );
?>
```


Ahora vamos a intentar hacer algo más interesante, nos conectaremos al servidor MySQL, ejecutaremos una consulta MySQL e imprimiremos los resultados como HTML. De nuevo, necesitaremos la contraseña del usuario root para MySQL. Guarde este archivo como /var/www/db.php:

```
<?php
$link = mysql_connect("localhost", "root", "newmysqlpassword");
if (!$link) {
   echo "Can't connect to database. Drat.\n";
   exit( );
}
$result = mysql_query("show databases");
if (!$result) {
   echo "Arggh, a database error: ", mysql_error( );
   exit( );
}
# print_r prints all of a variable's contents
while ($row = mysql_fetch_assoc($result))
   print_r($row);
?>
```

Introduzca la URL http://server1.centralsoft.org/db.php en su navegador y podrá ver:

```
Array ( [Database] => mysql ) Array ( [Database] => test )
```

Si hubiese usado el mismo comando SQL desde el cliente de línea de comandos, habría obtenido los mismos resultados (dos bases de datos, llamadas mysql y test), pero con un formato diferente:

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2996 to server version: 4.0.24_Debian-
10sarge2-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+----------+
| Database |
+----------+
| mysql    |
| test     |
+----------+
2 rows in set (0.00 sec)
```

Directivas específicas del módulo PHP

Las directivas PHP deben colocarse en su propio archivo de configuración PHP (/etc/php4/apache2/php.ini) o en el archivo de configuración de Apache.

Normalmente, no tendrá que manipularlas, a menos que instale un módulo PHP de extensión o quiera modificar dónde se deben buscar las librerías o configurar algunas opciones (como el modo seguro). Los módulos Apache normales tienen archivos de configuración con una extensión .conf, ubicados en el directorio /etc/apache2/mods-enabled.

Hosts Virtuales

Aunque podría tener un servidor Apache dedicado para un único sitio, probablemente querrá manejar más de un sitio. Apache llama a esta característica "hosts virtuales", y hay más de una manera para especificarlos. Cuando un cliente Web contacta con un servidor Web vía http, envía la dirección IP de destino y (en HTTP 1.1, el protocolo Web estándar actual) el nombre de un servidor de esa dirección.

En la configuración por defecto de Apache, no hay host virtuales independientes. Apache servirá las páginas Web sin importar cuántos nombres tenga el servidor, y todos los nombres de dominio comparten la misma configuración.

En los siguientes ejemplos, supondremos que necesita alojar cada sitio en su propio directorio en /var/www/vhosts.

Hosts virtuales basados en IP

Si tiene más de una dirección IP en su servidor y quiere dedicar ciertas direcciones a ciertos sitios, debe usar host virtuales basados en IP (o basados en dirección):

```
<VirtualHost 192.168.6.1>
   ServerName "www1"
   DocumentRoot "/var/www/vhosts/www1.example.com"
</VirtualHost>
<VirtualHost 192.168.6.2>
   ServerName "www2"
   DocumentRoot "/var/www/vhosts/www2.example.com"
</VirtualHost>
```

Esto era lo más común en los primeros días de la Web, debido a que HTTP 1.0 no tenía forma de especificar qué servidor quería consultar en esa dirección, ahora, con HTTP 1.1, el alojamiento basado en nombres es más popular.

Hosts virtuales basados en nombres

Con este método, la directiva NameVirtualHost define qué direcciones pueden ser hosts virtuales; * significa cualquier nombre o dirección de este nombre, incluyendo localhost, 127.0.0.1, www.centralsoft.org, www2.centralsoft.org, u otros. Las directivas ServerName asociadas con el nombre del servidor

indicado en la petición del navegador con el directorio donde se almacenan los archivos que se servirán:

```
# Accept any site name on any port:
NameVirtualHost *
<VirtualHost *>
   ServerName www1.example.com
   DocumentRoot "/var/www/vhosts/www1.example.com"
</VirtualHost>
<VirtualHost *>
   ServerName www2.example.com
   # A virtual host can have multiple names:
   ServerAlias backup.example.com
   DocumentRoot "/var/www/vhosts/www2.example.com"
</VirtualHost>
```

mod_vhost_alias

Si quiere administrar múltiples equipos sin necesidad de especificar los nombres de cada uno en los archivos de configuración, puede activar el módulo mod_vhost_alias de Apache:

```
# a2enmod vhost_alias
```

y configurar los nombres que se servirán en el archivo designado. El elemento vhost_alias del comando anterior sustituye al archivo /etc/apache2/mods-enabled/vhost_alias.conf. El contenido de ejemplo es:

```
UseCanonicalName Off
VirtualDocumentRoot /var/www/vhosts/%0
```

La directiva VirtualDocumentRoot es muy flexible. El %0 especificado aquí sustituye al nombre entero del sitio (server1.centralsoft.org). Podríamos haber usado %2 para obtener la segunda parte de la izquierda (centralsoft), %-2 para la segunda parte de la derecha (también centralsoft), %2+ para la segunda parte del nombre (centralsoft.org), etc. Estas alternativas son útiles si tiene muchos hosts virtuales. Si siempre tiene el mismo nombre base de dominio, como centralsoft.org, y los sitios se llaman www1.centralsoft.org, www2.centralsoft.org, etc., podría usar %1 para obtener los directorios /var/www/vhosts/www1, /var/www/vhosts/www2, etc.

Por ahora, solo use %0 para obtener el nombre completo y cree un directorio para cada host virtual:

```
# cd /var/www/vhosts
# mkdir www1.centralsoft.org
# echo "test www1.centralsoft.org" > www1.centralsoft.org/index.html
# mkdir www2.centralsoft.org
# echo "test www2.centralsoft.org" > www2.centralsoft.org/index.html
```

Luego reinicie Apache:

```
# /etc/init.d/apache2 reload
```

Si tiene registros DNS que apuntan a `www1.centralsoft.org` y `www2.centralsoft.org` desde su servidor, puede poner en el servidor `http://www1.centralsoft.org/index.html` y `http://www2.centralsoft.org` para ver los contenidos de los archivos `index.html` de prueba que acaba de hacer.

Archivos log

Apache escribe archivos ASCII de log de dos tipos: acceso (peticiones que se hacen al servidor) y error (errores que ocurren durante las peticiones). Puede controlar qué se escribe en estos archivos, dependiendo de lo que quiera saber sobre los visitantes de su sitio, cuánto espacio tenga en disco (los logs pueden hacerse muy grandes) y qué herramientas de análisis de logs quiera usar.

Un mensaje de acceso típico (divido en varias líneas para que quepa en la página) es:

```
192.168.0.1 - - [22/Sep/2006:15:04:05 -0400] "GET / HTTP/1.1"
200 580 "-" "Mozilla/5.0 (Windows; U; Windows NT 5.0; en-US;
rv:1.8.0.7) Gecko/20060909 Firefox/1.5.0.7"
```

Un mensaje de error típico es:

```
[Fri Sep 29 10:13:11 2006] [error]
[client www.centralsoft.org]
File does not exist: /var/www/index.html
```

Los logs por defecto son `/var/log/apache2/access.log` y `/var/log/apache2/error.log`.

División y rotación de logs

La configuración por defecto de Apache incluye una tarea diaria que hace rotar los logs de acceso y de errores. La rotación consiste en:

1. Renombrar `access.log` a `access.log.1` y `error.log` a `error.log.1`.
2. Incrementar el número de la extensión de logs rotados anteriormente (por ejemplo, `access.log.1` se incrementa a `Access.log.2`).
3. Borra `access.log.7` y `error.log.7`.
4. Crea un nuevo `access.log` y un `error.log`.

Por defecto, todos sus hosts virtuales comparten los mismos logs de acceso y de error. Si tiene más de un host, no obstante, será conveniente dividir los logs para ofrecer análisis diferentes para cada uno.

Apache tiene dos formatos estándar para archivos logs de acceso: común y combinado. Encontrará sus respectivas definiciones en el archivo de configuración maestro de Apache, `/etc/apache2/apache.conf`:

```
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\""
combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
```

Todos los % en los archivos de configuración de Apache representan variables; por ejemplo, %h significa nombre de equipo. El formato combinado es el formato común pero incluye el visitante y el agente de usuario (navegador). Desgraciadamente, ninguno de los dos formatos incluye el nombre del host virtual (una variable %) que necesita para dividir el log por equipos. Por tanto, si quiere hacer esto último, tendrá que definir un nuevo tipo de log.

En lugar de ir dejando huellas por todos los archivos de configuración de Apache, continúe haciendo los cambios en el sitio como lo hemos hecho hasta ahora (`/etc/apache2/sites-enabled/000-default`). Ponga estas líneas encima de sus directivas VirtualHost:

```
# Define a new virtual host common log format:
LogFormat "%v %h %l %u %t \"%r\" %s %b" vcommon
```

Dividiendo los logs con vlogger

Se estará preguntando si dividir la información de los logs en los archivos de Apache a medida que se va accediendo, o dividir el archivo de acceso una vez al día con una utilidad para dividir los archivos log de Apache. Preferimos la primera opción, debido a que coloca las líneas en el log adecuado inmediatamente, y no necesitamos escribir tareas cron. Un buen programa es vlogger. Apache le permite conectar el log con algunos programas externos, que es justo lo que queremos. Añada toda esta información debajo de la línea LogFormat que introdujo previamente:

```
# Split log on the fly into virtual host directories
# under /var/log/apache2:
CustomLog "| /usr/sbin/vlogger -s access.log /var/log/apache2" vcommon
```

Ya que vlogger no forma parte del paquete estándar de Debian, instálelo:

```
# apt-get install vlogger
```

Luego reinicie Apache:

```
# /etc/init.d/apache2 restart
```

vlogger creará un archivo en el directorio /var/log/apache2 para cada host virtual que ha definido. Se creará un log de acceso diario con marcas de tiempo, y con un enlace simbólico al archivo access.log más reciente:

```
# cd /var/log/apache2/www1.example.com
# ls -l
total 4
-rw-r--r-- 1 root root 984 Aug 3 23:19 08032006-access.log
lrwxrwxrwx 1 root root 19 Aug 3 23:19 access.log -> 08032006-access.log
```

Analizando logs con Webalizer

Hay muchos analizadores de logs libres y comerciales: Nosotros pensamos que Webalizer es una buena opción porque es fácil de instalar y genera una salida muy útil:

Introduzca:

```
# apt-get install webalizer
...
Which directory should webalizer put the output in?
/var/www/webalizer
Enter the title of the reports webalizer will generate.
Usage Statistics for server1.centralsoft.org
What is the filename of the rotated webserver log?
/var/log/apache2/access.log.1
```

Acceda a él a través del URL http://server1.centralsoft.org/webalizer.

Al día siguiente (después de que se ejecute la tarea cron de Webalizer /etc/cron.daily/webalizer) podrá ver páginas con tablas que describen el acceso a su servidor Web. No necesita editar el archivo de configuración (/etc/webalizer.conf), a menos que quiera cambiar las opciones que proporcionó durante la instalación.

Nota: Los spammers tienen formas de manipular los logs Web, por lo que es una buena práctica restringir el acceso a las páginas de salida de Webalizer.

Encriptación SSL/TLS

Willie Sutton dijo que robaba bancos porque era "donde estaba el dinero". Los ataques en Internet se están incrementando, sobre todo los ataques a nivel de aplicación, por la misma razón. Encriptar los datos sensibles como los números de tarjeta de crédito y las contraseñas es esencial.

Cuando solicita una página desde un servidor Web con el prefijo http://, todos los datos que circulan desde el servidor hasta el navegador están sin cifrar. Cualquiera con acceso a las redes que intervienen en la comunicación puede capturar los contenidos. Piense que el acceso Web (como el correo electrónico estándar) es una postal en lugar de una carta.

El estándar *Secure Sockets Layer* fue desarrollado para encriptar el tráfico Web, y ha sido un factor decisivo en la explosión de sitios comerciales y del comercio electrónico en la Web. Apache tiene la capacidad de encriptar el tráfico Web con SSL, que con ligeras modificaciones se conoce como la capa *Transport Layer Security*. Este cifrado se obtiene cuando accede a un sitio con el prefijo https://. Imagine que el tráfico Web cifrado es como un sobre sellado.

Vamos a configurar SSL para Apache. Edite /etc/apache2/ports.conf y añada esta línea:

```
Listen 443
```

Luego active el módulo SSL de Apache e indique a Apache que debe usarlo:

```
# a2enmod ssl
```

El módulo ssl está instalado; ejecute /etc/init.d/apache2 force-reload para activarlo.

```
# /etc/init.d/apache2 force-reload
```

Ahora intente acceder a su página de inicio con https:// URL (por ejemplo, https://server1.centralsoft.org).

Para que SSL funcione, su servidor necesita un certificado. Este certificado es un archivo encriptado que prueba que el usuario es quien dice ser. ¿Cómo sabe el navegador en quién debe confiar? Los navegadores Web disponen de listas de autoridades de certificación (CA). La cadena de comandos para ver estas listas es:

```
Firefox 2.0
   Herramientas ⟶ Avanzadas ⟶ Cifrado ⟶ Ver certificados ⟶ Autoridades

Internet Explorer 6.0
   Herramientas ⟶ Opciones de Internet ⟶ Contenido ⟶ Certificados ⟶
   Certificado de confianza raíz
```

Los CA son empresas que venden a las organizaciones un certificado y se encargan del trabajo legal para verificar su identidad. Los sitios Web comerciales casi siempre usan CA comerciales, porque el navegador suele aceptar los certificados que expenden. De manera alternativa, usted puede ser su propio CA y crear

un certificado propio. Esto funciona tanto con SSL como con certificados comerciales, pero el navegador Web le preguntará al usuario si desea o no aceptar el certificado. Los certificados propios son comunes en proyectos de código abierto pequeño y durante las pruebas de proyectos grandes.

Soporte para suEXEC

Apache puede servir múltiples sitios al mismo tiempo, pero hay sitios que tienen páginas distintas, scripts CGI, usuarios, etc. Debido a que Apache se ejecuta con los permisos de un determinado usuario y grupo (por defecto www-data), el usuario puede escribir y leer el contenido de todos estos sitios. Pero nosotros queremos asegurarnos que sólo los miembros de un sitio determinado pueden ejecutar los programas de ese sitio y acceder a los datos de ese sitio. Como de costumbre, hay más de una forma de hacer esto, usando varias combinaciones de Apache, PHP y otras herramientas.

Un método popular es usar suEXEC, un programa que se ejecuta con permisos de root y hace que los programas CGI se ejecuten con el ID de usuario y de grupo de un usuario específico, no el usuario y el grupo del usuario que ejecuta el servidor Apache. Por ejemplo, al usar nuestro segundo host virtual `www2.example.com`, con la cuenta de usuario `www-user2` y el grupo www-group2, cambiaremos los permisos del host virtual especificando:

```
<VirtualHost www2.example.com>
   SuExecUserGroup www-user2 www-group2
</VirtualHost>
```

Rendimiento

Nuestro principal objetivo es instalar y configurar nuestro servidor Web de manera correcta y segura. Además de esto, queremos asegurarnos que puede soportar la carga de trabajo que suponen nuestros sitios Web. La Web tiene muchas partes dinámicas, es muy fácil que estas partes produzcan el bloqueo del sistema. Para evaluar el rendimiento de nuestro sistema, vamos a usar herramientas especializadas que simulan cientos de usuarios activos (que por otra parte es mucho más barato que pagar a cientos de usuarios para hacer las pruebas).

Nota: Apache puede ejecutarse con diferentes versiones, llamadas modelos. La instalación por defecto de Debian es el modelo tenedor, en el cual varios procesos Apache atienden las peticiones. Este suele ser el modelo que mejor funciona bajo Linux.

Como mínimo se necesita un archivo HTML estático para evaluar el rendimiento. Cree un archivo llamado `/var/www/bench.html`. Debería ser aproximadamente del tamaño que cabe esperar para una página de su sitio. Puede impresionar a sus amigos generando texto en Latin desde el sitio `http://www.lipsum.com` y pegándolo en `bench.html`. El programa que evalúa el rendimiento es ab, perteneciente al paquete apache2-utils, y debería instalarse junto con Apache. Hagamos 1000 peticiones distintas para el mismo archivo, con concurrencia 5 (5 peticiones simultáneas):

```
# ab -n 1000 -c 5 http://server1.centralsoft.org/bench.html
This is ApacheBench, Version 2.0.41-dev <$Revision$> apache-2.0
Copyright (c) 1996 Adam Twiss, Zeus Technology Ltd,
http://www.zeustech.net/
Copyright (c) 1998-2002 The Apache Software Foundation,
http://www.apache.org/

Benchmarking server1.centralsoft.org (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Finished 1000 requests

Server Software:        Apache/2.0.54
Server Hostname:        server1.centralsoft.org
Server Port:            80

Document Path:          /bench.html
Document Length:        1090 bytes

Concurrency Level:      5
Time taken for tests:   2.799386 seconds

Complete requests:      1000
Failed requests:        0
Write errors:           0
Non-2xx responses:      1000
Total transferred:      1425000 bytes
HTML transferred:       1090000 bytes
Requests per second:    357.22 [#/sec] (mean)
Time per request:       13.997 [ms] (mean)
Time per request:       2.799 [ms] (mean, across all concurrent requests)
Transfer rate:          496.89 [Kbytes/sec] received

Connection Times (ms)
```

```
                min  mean[+/-sd]  median   max
Connect:          0     0   0.1      0       3
Processing:       6    11   2.2      1      22
Waiting:          5    10   2.3     11      18
Total:            6    11   2.2     11      22

Percentage of the requests served within a certain time (ms)
50%   11
66%   12
75%   13
80%   13
90%   14
95%   14
98%   15
99%   16
100%  22 (longest request)
```

Normalmente, la gente quiere ver las peticiones por segundo o su inverso, el tiempo por petición. Estos números le indicarán qué debe mejorar en relación con el hardware del servidor y con la configuración de Apache.

Instalando y administrando Drupal

Ahora ya tenemos Apache, PHP y MySQL ejecutándose, instalemos un paquete para usarlos. Desgraciadamente, no vamos a pagar por un producto comercial, por lo que escogeremos algún producto de código abierto, que represente un ejemplo típico de software real y que sea bastante útil. Vamos a visitar por ejemplo `http://www.drupal.org`:

Drupal es un software que permite a un individuo o a una comunidad de usuario publicar, gestionar y organizar una variedad de contenidos en un sitio Web.

Entre estas funcionalidades se incluyen foros, gestión de documentos, galerías, grupos de noticias y otras formas de colaboración basadas en Web.

Las siguientes dos secciones describen los dos métodos de instalación de Drupal:

- apt-get: Es lo más fácil, así es que inténtelo en primer lugar. Sin embargo, nosotros hemos tenido problemas con los paquetes Debian de Drupal.
- Desde las fuentes: Supone más trabajo, pero puede ver qué está ocurriendo, inténtelo de esta forma si el método apt-get falla.

Instalando Drupal con apt-get

La forma más fácil de instalar Drupal es con apt-get. Puede ir al sitio Web de Drupal y buscar un paquete listo para descargar o puede ver mediante apt-cache si está en el repositorio de Debian:


```
# apt-cache search drupal
drupal - fully-featured content management/discussion engine
drupal-theme-marvinclassic - "Marvin Classic" theme for Drupal
drupal-theme-unconed - "UnConeD" theme for Drupal
```

El primero es el que nosotros queremos, por lo que vamos a instalarlo de la siguiente manera:

```
# apt-get install drupal
```

El proceso de instalación indica que necesita varios paquetes que no tiene, consígalos e instálelos. Luego se le preguntará por la configuración de Drupal a través de una secuencia de menús. Use el tabulador para moverse entre las opciones, la barra espaciadora para seleccionar una opción o **Intro** para ir a la siguiente página. Solamente incluiremos la última línea de cada pantalla aquí, eso sí, con la respuesta recomendada:

```
Automatically create Drupal database?
Yes

Run database update script?
Yes

Database engine to be used with Drupal
MySQL

Database server for Drupal's database
localhost

Database server administrator user name on host localhost
root

Password for database server administrator root on localhost
newmysqlpassword

Drupal database name
Drupal

Remove Drupal database when the package is renoved?
No

Remove former database backups when the package is removed?
Yes

Web server(s) that should be configured automatically
[ ] apache
[ ] apache-ssl
[ ] apache-perl
[*] apache2
```

La instalación copiará los archivos de programa, creará una base de datos MySQL y creará un archivo de configuración Apache (/etc/apache2/conf.d/drupal.conf):

```
Alias /drupal /usr/share/drupal
<Directory /usr/share/drupal/>
   Options +FollowSymLinks
   AllowOverride All
   order allow,deny
   allow from all
</Directory>
```

Si aparece un mensaje parecido a este:

```
An override for "/var/lib/drupal/files" already exists, but -force
specified so lets ignore it.
```

puede quebrarse la cabeza buscando una solución, o instalarlo desde las fuentes. Si todo ha ido bien, sáltese la siguiente sección.

Instalando Drupal desde las fuentes

Descargue la última distribución y vaya al directorio raíz de su servidor Web:

```
# wget http://ftp.osuosl.org/pub/drupal/files/projects/drupal-4.7.3.tar.gz
# tar xvzf drupal-4.7.3.tar.gz
# mv drupal-4.7.3 /var/www/drupal
# cd /var/www/drupal
```

Seguiremos los pasos indicados en INSTALL.txt y en INSTALL.mysql.txt. Crearemos la base de datos Drupal (la llamaremos drupal), un usuario administrador (también drupal, puesto que no tenemos mucha imaginación) y una contraseña (por favor, no use drupal también):

```
# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 37 to server version: 4.0.24_Debian-10sarge2-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database drupal;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP,
-> INDEX, ALTER, CREATE TEMPORARY TABLES, LOCK TABLES
-> on drupal.* to
-> "drupal"@"localhost" identified by "drupalpw";
```



```
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> quit;
Bye
```

Luego, cargue la definición de la base de datos de Drupal en MySQL:

```
# mysql -u root -p drupal < database/database.4.0.mysql
Enter password:
#
```

Después edite el archivo default/config.php y cambie la línea:

```
$db_url = 'mysql://username:password@localhost/databasename';
```

por:

```
$db_url = 'mysql://drupal:drupalpw@localhost/drupal';
```

Configurando Drupal

En su navegador Web, vaya a http://server1.centralsoft.org/drupal. La primera página (en la versión que hemos probado) dice:

```
Welcome to your new Drupal website!
Please follow these steps to set up and start using your website:
Create your administrator account
To begin, create the first account. This account will have full
administration rights
and will allow you to configure your website.
```

Haga clic en el enlace Crear la primera cuenta. En esta segunda página, teclee el nombre deseado para su cuenta (o su nombre completo) en el campo de texto Username y su correo en el campo E-Mail. Luego presione el botón **Crear una nueva cuenta**. Será redirigido a la primera página, donde en la parte de arriba podrá leer:

Compruebe su correo electrónico para ver si le ha llegado la contraseña generada y autentifíquese en Drupal en el área "User login". Será redirigido a una página para establecer una contraseña permanente. Después de esto, puede ir a la página de inicio, donde verá las siguientes opciones:

1. **Cree su cuenta de administrador.**
   Para empezar, cree la primera cuenta. Esta cuenta tiene plenos derechos de administración por lo que le permitirá configurar el sitio Web.

2. **Configure su sitio Web.**
   Una vez en el sistema, visite la sección de administración, donde puede personalizar y configurar todos los aspectos de su sitio Web.
3. **Active la funcionalidad adicional.**
   Luego, visite la lista de módulos y actívelos en función de sus necesidades específicas. Puede encontrar módulos adicionales en la sección de descarga de módulos de Drupal.
4. **Personalice el diseño de su sitio Web.**
   Para cambiar la apariencia de su sitio Web, visite la sección de temas. Deberá elegir uno de los temas incluidos o descargar temas adicionales de la sección de descarga de temas de Drupal.
5. **Comience a publicar contenido.**
   Finalmente, puede crear contenido para su sitio Web. Este mensaje desaparecerá una vez que haya publicado por primera vez.
   Para más información, vaya a la sección de ayuda, o a los libros en línea de Drupal. También puede enviar una pregunta al foro o usar un amplio rango de opciones de soporte.

Puesto que ya ha creado la primera cuenta (la de administrador). Ahora ya puede probar todas las funcionalidades.

Resolución de problemas

Si le gusta diagnosticar problemas, probablemente le gustará la Web. Hay muchas cosas que fallan, en muchos lugares y de muchas formas, por lo que estaría ocupado durante muchos años. Echemos un vistazo a algunos problemas Web clásicos. (Los mensajes de error del navegador que usa Firefox, aunque los de Internet Explorer son similares.)

La página Web no aparece en el navegador

Supongamos que el directorio raíz es `/var/www`, su archivo es test.html y su servidor es `server1.centralsoft.org`. Cuando usa un navegador Web externo `http://server1.centralsoft.org/test.html`, obtiene una página de error en la ventana del navegador. Un mensaje de error del navegador como "Servidor no encontrado" implica un problema DNS. Primero, asegúrese de que `server1.centralsoft.org` tiene entradas DNS en un servidor de nombres público:

```
# dig server1.centralsoft.org
...
;; ANSWER SECTION:
```



```
server1.centralsoft.org.     106489    IN    A     192.0.34.166
...
```

Luego vea si el servidor puede alcanzarse desde Internet. Si su cortafuegos permite pings, sondee si el servidor responde a la los pings desde fuera:

```
# ping server1.centralsoft.org
PING server1.centralsoft.org (192.0.34.166) 56(84) bytes of data.
64 bytes from server1.centralsoft.org (192.0.34.166): icmp_seq=1 ttl=49
time=81.6 ms
```

Compruebe que el puerto 80 está abierto y no está bloqueado. Pruebe con nmap desde una máquina externa:

```
# nmap -P0 -p 80 server1.centralsoft.org

Starting nmap 3.81 ( http://www.insecure.org/nmap/ ) at 2006-07-25
23:50 CDT
Interesting ports on server1.centralsoft.org (192.0.34.166):
PORT  STATE SERVICE
80/tcp open http

Nmap finished: 1 IP address (1 host up) scanned in 0.186 seconds
```

Si no tiene nmap, use telnet para conectarse al puerto 80 y haga la petición HTTP más simple posible:

```
# telnet server1.centralsoft.org 80
Trying 192.0.34.166...
Connected to server1.centralsoft.org.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Wed, 26 Jul 2006 04:52:13 GMT
Server: Apache/2.0.54 (Fedora)
Last-Modified: Tue, 15 Nov 2005 13:24:10 GMT
ETag: "63ffd-1b6-80bfd280"
Accept-Ranges: bytes
Content-Length: 438
Connection: close
Content-Type: text/html; charset=UTF-8

Connection closed by foreign host.
```

Si no funciona, asegúrese de que esta línea está en `/etc/apache2/ports.conf`:

```
Listen 80
```

y también, que puede ver algo muy parecido a esto en el puerto 80, como lo siguiente:

```
# lsof -i :80
COMMAND   PID     USER     FD   TYPE  DEVICE SIZE  NODE NAME
apache2  10678  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  10679  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  10680  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20188      root   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20190  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20191  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20192  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20194  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20197  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20198  www-data   3u   IPv6  300791       TCP *:www (LISTEN)
apache2  20199 www-data 3u IPv6  300791   TCP *:www (LISTEN)
```

Si no ve apache2 en esta salida, compruebe que Apache se está ejecutando:

```
# ps -efl | grep apache2
```

Si la salida contiene líneas como esta:

```
5 S root    7692     1  0  76    0 - 2991 415244 Jul16 ?    00:00:00
/usr/sbin/apache2 -k start -DSSL
```

Apache se está ejecutando. Si no es así, reinícielo:

```
# /etc/init.d/apache2 start
```

Luego ejecute el comando ps de nuevo. Si Apache sigue sin aparecer, mire el log de errores:

```
# tail -f /var/log/apache2/error.log
```

Si no tiene permisos para ver este archivo, definitivamente hoy no es su día. Si el log de errores está vacío, quizá tenga los permisos equivocados. Confirme que el directorio `/var/log/apache2` y `/var/log/apache2/error.logfile` existen:

```
# ls -l /var/log/apache2
total 84
-rw-r----- 1 root adm 31923 Jul 25 23:09 access.log
-rw-r----- 1 root adm 32974 Jul 22 20:50 access.log.1
-rw-r----- 1 root adm 379 Jul 23 06:25 access.log.2.gz
-rw-r----- 1 root adm 1969 Jul 25 23:09 error.log
-rw-r----- 1 root adm 1492 Jul 23 06:25 error.log.1
-rw-r----- 1 root adm 306 Jul 23 06:25 error.log.2.gz
```

Si el log de errores muestra otra información más antigua, probablemente quede poco espacio en disco. Es sorprendente el número de veces que olvidamos comprobar esto antes de investigar cosas más esotéricas como los cortafuegos. Teclee:


```
# df
Filesystem     1K-blocks      Used     Available   Use% Mounted on
/dev/hda1      193406200      455292   183126360   1% /
tmpfs          453368         0        453368      0% /dev/shm
```

Si ha usado una directiva User o Group diferente en la configuración de Apache, compruebe que el usuario y el grupo existen:

```
# id www-data
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Si el navegador devolvió un mensaje de error de Apache, todavía tiene campo para seguir investigando. Si lo que se muestra es:

```
Not Found
The requested URL /wrong.html was not found on this server.
```

la URL probablemente no existe. Si puede ver:

```
Forbidden
You don't have permission to access /permissions.html on this server.
```

el archivo está ahí, pero el usuario Apache no puede leerlo.

```
# cd /var/www
# ls -l permissions.html
-rw------- 1 root root 0 Jul 26 00:01 permissions.html
```

Los problemas de permisos pueden solucionarse cambiando el propietario del archivo al proceso que ejecuta Apache.

Los Hosts Virtuales no funcionan:

Use

```
# apache2ctl -S
```

Para comprobar las directivas de host virtual.

SSI no funciona

Si ve líneas como estas en el log de error (`/var/log/apache2/error.log`):

```
[error] an unknown filter was not added: INCLUDES
```

es que no activó `mod_include`. Ejecute el comando:

```
# a2enmod include
```

Un programa CGI no se ejecuta

Si no consigue que un programa CGI se ejecute, pruebe las siguientes soluciones:

- ¿Se ha activado el CGI usando alguno de los métodos ya descritos anteriormente?
- ¿Está el programa CGI es un directorio CGI como /var/cgi-bin o tiene un sufijo como .php?
- ¿Se puede leer el archivo? Si no, use chmod.
- ¿Qué dice el log de errores de Apache?
- ¿Qué dice el log de error del sistema, /var/log/messages?

SSL no funciona

Compruebe que tiene activado el módulo SSL de Apache (a2enmod ssl) e indique a Apache que escuche en el puerto 443 en /etc/apache2/ports.conf:

```
Listen 443
```

Si la directiva no estuviera ahí, añádala y reinicie Apache. Luego intente acceder a esta URL desde su navegador: https://server1.centralsoft.org. Si sigue sin funcionar, puede que el puerto 443 esté bloqueado por un cortafuegos. Puede comprobarlo con nmap:

```
# nmap -P0 -p 443 server1.centralsoft.org

Starting nmap 3.70 ( http://www.insecure.org/nmap/ ) at 2006-08-01
22:38 CDT
Interesting ports on ... (...):
PORT  STATE SERVICE
443/tcp open   https

Nmap run completed -- 1 IP address (1 host up) scanned in 0.254 seconds
```

Capítulo 7

Clusters de carga balanceada



Hace más de 10 años, las personas descubrieron que podían conectar múltiples máquinas baratas para realizar tareas de computación que en condiciones normales requerirían un mainframe o un supercomputador. El clúster Beowulf de la NASA fue uno de los primeros ejemplos y aún hoy se sigue utilizando (http://www.beowulf.org). En la Wikipedia (http://en.wikipedia.org/wiki/Computer_cluster) están recogidas algunas de las características principales de un clúster: un clúster es un grupo débilmente acoplado de ordenadores que trabajan juntos y que en muchos aspectos puede verse como un único ordenador. Los clusters a menudo, pero no siempre, están conectados mediantes redes de área local de alta velocidad. Los clusters se suelen desplegar para mejorar la velocidad o la fiabilidad que ofrece un único ordenador, puesto que ofrecen una relación efectividad/coste mucho mejor que un único ordenador con las mismas prestaciones.

Los clusters son una buena solución cuando lo que busca es mejorar la velocidad, fiabilidad y escalabilidad a un precio razonable. Amazon, Yahoo! y Google basan sus negocios en miles de servidores configurados como clusters. Es más barato y más fácil ampliar su negocio horizontalmente (añadiendo más servidores) que verticalmente (comprando máquinas más caras). Hay muchas soluciones Linux para clusters, tanto libres como comerciales. En este capítulo describiremos algunos clusters basados en el Linux Virtual Server (http://www.linuxvirtual-server.org) que es libre. Le mostraremos cómo combinar tres ordenadores en un clúster de carga balanceada para el servidor Apache. También describiremos sus capacidades y, finalmente, las alternativas que existen. No pretendemos cubrir clusters de alto rendimiento, computación en grid, paralelismo o computación distribuida; en estos campos, tanto el hardware como el software están mucho más especializados (por ejemplo, para predicción meteorológica o renderizado de gráficos).

Balanceo de carga y alta disponibilidad

El balanceo de carga (LB) ofrece escalabilidad: la distribución de las peticiones en varios servidores. LB consiste en el reenvío de paquetes y en el conocimiento del servicio cuya carga va a balancearse (en este capítulo, HTTP). Se basa en un monitor externo que recoge las estadísticas de carga de los servidores físicos para decidir dónde se deben enviar los paquetes.

La alta disponibilidad (HA) ofrece fiabilidad: mantiene los servicios ejecutándose. Se basa en servidores redundantes, intercambio de mensajes del tipo "Estoy vivo", y un procedimiento para que en caso de fallos se sustituya el servidor donde se produjo el error por otro.

En este capítulo, vamos a concentrarnos en LB, puesto que los administradores tendrán que enfrentarse a ello más a menudo. Para un sitio que pase a ser sensible dentro de una organización, la HA también es necesaria. Al final de este capítulo, ofreceremos varios enlaces útiles para configurar sistemas que combinan balanceo de carga y alta disponibilidad.

El ejemplo de balanceo de carga que vamos a usar en este capítulo es un ejemplo simple que consiste en tres direcciones públicas y una virtual. Todas ellas se listan en la tabla 7.1.

Tabla 7.1. Direcciones y roles de los servidores en nuestro clúster.

Nombre	Dirección IP	Descripción
Lb	70.253.158.44	**Balanceador de carga—servicio con dirección pública**
web1	70.253.158.41	**Primer servidor web—una de las IP reales (RIP)**
web2	70.253.158.45	**Segundo servidor web—otra RIP**
(VIP)	70.253.158.42	**IP virtual (VIP) compartida por lb, web1 y web2, además de sus direcciones IP reales**

VIP es la dirección que el balanceador de carga expone a los clientes, desde donde se distribuirán las peticiones a los servidores Web.

Software para balanceo de carga

La forma más simple de balanceo de carga es un round-robin de DNS, donde múltiples registros A se definen con el mismo nombre, el resultado es que los servidores esperan su turno para responder a las peticiones. Esto no funcionaría bien si un servidor falla, además, no tiene en cuenta las necesidades específicas

del servicio. Con http, por ejemplo, si se necesita mantener una sesión de datos con autentificación o cookies, hay que asegurarse de que el cliente siempre se conecta al mismo servidor. Para cumplir estas necesidades, usaremos dos herramientas un poco sofisticadas:

- Servidor de IP virtual (IPVS): Un módulo balanceador de carga de nivel de transporte (TCP) que ahora es un componente estándar de Linux.
- ldirectord: Una utilidad que monitoriza el estado de los servidores físicos. Las instrucciones de instalación están basadas en la distribución de Linux Debian 3.1 (Sarge).

IPVS en el balanceador de carga

Ya que IPVS está ya en el núcleo de Linux, no necesitamos instalar software, pero sí configurarlo.

En lb, añada estas líneas a `/etc/modules`.

```
ip_vs_dh
ip_vs_ftp
ip_vs
ip_vs_lblc
ip_vs_lblcr
ip_vs_lc
ip_vs_nq
ip_vs_rr
ip_vs_sed
ip_vs_sh
ip_vs_wlc
ip_vs_wrr
```

Luego cargue los módulos en el núcleo:

```
# modprobe ip_vs_dh
# modprobe ip_vs_ftp
# modprobe ip_vs
# modprobe ip_vs_lblc
# modprobe ip_vs_lblcr
# modprobe ip_vs_lc
# modprobe ip_vs_nq
# modprobe ip_vs_rr
# modprobe ip_vs_sed
# modprobe ip_vs_sh
# modprobe ip_vs_wlc
# modprobe ip_vs_wrr
```

Para activar el reenvío de paquetes en el núcleo de Linux de lb, edite el archivo `/etc/sysctl.conf` y añada esta línea:

```
net.ipv4.ip_forward = 1
```

Para cargar esta opción en el núcleo:

```
# sysctl -p
net.ipv4.ip_forward = 1
```

ldirectord

Aunque podríamos obtener ldirectord de manera independiente, lo obtendremos como parte del paquete Ultra Monkey, que además incluye software para HA. Debido a que Ultra Monkey no forma parte de la distribución Debian estándar, necesitará añadir dos líneas a su repositorio de archivos de Debian (`/etc/apt/sources.list`) en la máquina lb:

```
deb http://www.ultramonkey.org/download/3/ sarge main
deb-src http://www.ultramonkey.org/download/3 sarge main
```

Luego, actualice el repositorio y obtenga el paquete:

```
# apt-get update
# apt-get install ultramonkey
```

El proceso de instalación le hará algunas preguntas:

```
Do you want to automatically load IPVS rules on boot?
No
Select a daemon method.
none
```

Nuestra configuración tendrá un servidor virtual (la dirección que ven los clientes, ejecutando ldirectord), que llamaremos director, y dos servidores reales (ejecutando Apache). Los servidores reales pueden conectarse con el director de tres formas:

- LVS-NAT

  Los servidores reales están en una subred NAT detrás del director que es el quien enruta las respuestas.
- LVS-DR

  Los servidores reales envían las respuestas directamente al cliente. Todas las máquinas están en la misma subred y pueden comunicarse mediante direcciones Ethernet. No es necesario que se pueda hacer ping desde fuera de la subred.
- LVS-TUN

  Los servidores reales pueden estar en una red distinta de la del director. Se comunican mediante técnicas de tunneling IP sobre IP (IPIP).

Vamos a usar DR porque es fácil, rápido y escalable. Con este método, designamos una VIP que comparten el balanceador de carga y los servidores reales. Esto provoca un problema: si todas las máquinas tienen la misma VIP, ¿cómo se asocia dicha VIP a una dirección MAC física? Esto se llama el problema ARP, debido a que los sistemas de la misma LAN usan el *Address Resolution Protocol* (ARP) para encontrarse, y ARP espera que cada sistema tenga una dirección IP única.

Muchas soluciones necesitan parches o módulos del núcleo que cambian junto con el núcleo de Linux. En la versión 2.6 y superiores, una solución muy popular es permitir que el balanceador de carga maneje el ARP de la VIP y, en los servidores reales, configurar la VIP como alias del bucle local. Por esta razón, los dispositivos locales no responden a las peticiones ARP.

Esta es la solución que adoptaremos. Configuraremos todos los servidores Web primero.

Configurando los servidores reales (Nodos Apache)

En cada servidor real (web1 y web2) hay que hacer lo siguiente:

1. Si el servidor Apache todavía no está instalado, instalarlo:

   ```
   # apt-get install apache2
   ```

   Si todavía no tiene los archivos de contenido para su sitio Web, puede hacerlo ahora o después de configurar el sistema de balanceo de carga.
2. Instale iproute (un paquete de Linux para trabajo en red con más funcionalidades que herramientas antiguas como ifconfig y route):

   ```
   # apt-get install iproute
   ```
3. Añada estas líneas a `/etc/sysctl.conf`:

   ```
   net.ipv4.conf.all.arp_ignore = 1
   net.ipv4.conf.eth0.arp_ignore = 1
   net.ipv4.conf.all.arp_announce = 2
   net.ipv4.conf.eth0.arp_announce = 2
   ```
4. Actualice los cambios en el núcleo:

   ```
   # sysctl -p
   net.ipv4.conf.all.arp_ignore = 1
   net.ipv4.conf.eth0.arp_ignore = 1
   net.ipv4.conf.all.arp_announce = 2
   net.ipv4.conf.eth0.arp_announce = 2
   ```
5. Suponiendo que su servidor real es Debian, edite el archivo `/etc/network/interfaces`, asociando la VIP (70.253.15.42) con el alias lo:0:

   ```
   auto lo:0
   iface lo:0 inet static
     address 70.253.15.42
     netmask 255.255.255.255
     pre-up sysctl -p > /dev/null
   ```

6. Active dicho alias:

```
# ifup lo:0
```

7. Cree el archivo /var/www/ldirector.html con el contenido:

```
¡Estoy vivo!
```

8. En web1:

```
# echo "I'm web1" > /var/www/which.html
```

9. En web2:

```
# echo "I'm web2" > /var/www/which.html
```

10. Inicie Apache, o reinícielo si ya se está ejecutando:

```
# /etc/init.d/apache2 restart
```

Los logs de acceso de Apache no deberían mostrar actividad, puesto que lb todavía no se ha comunicado con ellos.

Configurando el balanceador de carga

En lb, cree el archivo de configuración del balanceador de carga, /etc/ha.d/ldirectord.cf:

```
checktimeout=10
checkinterval=2
autoreload=no
logfile="local0"
quiescent=no
virtual=70.253.158.42:80
   real=70.253.158.41:80 gate
   real=70.253.158.45:80 gate
   service=http
   request="director.html"
   receive="I'm alive!"
   scheduler=rr
   protocol=tcp
   checktype=negotiate
```

Si quiescent fuera yes, un servidor real que fallara dejaría de contar para el balanceador, pero continuaría en la tabla LVS de routing; lo hemos puesto a no, por lo que los servidores caídos se eliminan de la tabla. El peso de un servidor refleja su capacidad en relación con los otros servidores. Para un esquema simple como el nuestro, todos los servidores operativos tienen un peso de 1, y los caídos un peso de 0. Si checktype es negotiate, el director hará una petición http a cada uno de los servidores reales de la URL solicitada, y verá si sus contenidos tienen el valor recibir. Si el valor fuera check, sólo se haría una comprobación TCP rápida, y las peticiones y las respuestas se ignorarían.

Se deberían haber creado archivos de inicio en /etc para ldirectord durante la instalación. Ultra Monkey también instaló Heartbeat, como no lo vamos a usar, lo desinstalamos:

```
# update-rc.d heartbeat remove
update-rc.d: /etc/init.d/heartbeat exists during rc.d purge
(use -f to force)
```

El balanceador de carga monitoriza el estado de los servidores Web, para lo que regularmente solicita el archivo que hemos especificado en ldirectord.cd(request="director.html").

Puesto que este servidor responderá las peticiones Web que se hagan a través de la dirección VIP (70.253.158.42), debemos indicárselo al servidor. Edite /etc/network/interfaces y añada estas líneas para crear el alias eth0:0:

```
auto eth0:0
iface eth0:0 inet static
   address 70.253.158.42
   netmask 255.255.255.248
# These should have the same values as for eth0:
   network ...
   broadcast ...
   gateway ...
```

Ahora active esta nueva dirección IP:

```
# ifup eth0:0
```

Finalmente, inicie los ldirectord en lb:

```
# /etc/init.d/ldirectord start
Starting ldirectord... success
```

Probando el sistema

Comprobemos si el balanceador de carga se está ejecutando en lb:

```
# ldirectord ldirectord.cf status
```

Debería ver algo parecido a esto:

```
ldirectord for /etc/ha.d/ldirectord.cf is running with pid:
1455
```

Si viera algo como esto:

```
ldirectord is stopped for /etc/ha.d/ldirectord.cf
```

habría un problema. Puede parar el director y reiniciarlo con la bandera de depuración -d y ver los errores que se están produciendo:

```
# /usr/sbin/ldirectord /etc/ha.d/ldirectord.cf stop
# /usr/sbin/ldirectord -d /etc/ha.d/ldirectord.cf start
DEBUG2: Running exec(/usr/sbin/ldirectord -d /etc/ha.d/ldirectord.cf start)
Running exec(/usr/sbin/ldirectord -d /etc/ha.d/ldirectord.cf start)
DEBUG2: Starting Linux Director v1.77.2.32 with pid: 12984
Starting Linux Director v1.77.2.32 with pid: 12984
DEBUG2: Running system(/sbin/ipvsadm -A -t 70.253.158.42:80 -s rr )
Running system(/sbin/ipvsadm -A -t 70.253.158.42:80 -s rr )
DEBUG2: Added virtual server: 70.253.158.42:80
Added virtual server: 70.253.158.42:80
DEBUG2: Disabled server=70.253.158.45
DEBUG2: Disabled server=70.253.158.41
DEBUG2: Checking negotiate: real
server=negotiate:http:tcp:70.253.158.41:80:::\/director\.html:I\'m\ alive\!
(virtual=tcp:70.253.158.42:80)
DEBUG2: check_http: url="http://70.253.158.41:80/director.html"
virtualhost="70.253.158.41"
LWP::UserAgent::new: ( )
LWP::UserAgent::request: ( )
LWP::UserAgent::send_request: GET http://70.253.158.41:80/director.html
LWP::UserAgent::_need_proxy: Not proxied
LWP::Protocol::http::request: ( )
LWP::Protocol::collect: read 11 bytes
LWP::UserAgent::request: Simple response: OK
45:80/director.html is up
```

La salida será más corta si checktype está definido como check.

Para curiosear, podemos ver los mensajes de bajo nivel del servidor de IP virtual:

```
# ipvsadm -L -n
IP Virtual Server version 1.2.0 (size=4096)
Prot LocalAddress:Port Scheduler Flags
-> RemoteAddress:Port     Forward Weight   ActiveConn  InActConn
TCP 70.253.158.42:80 rr
-> 70.253.158.45:80       Route    1        1            2
-> 70.253.158.41:80       Route    1        0            3
```

Esto muestra que nuestro primer servidor real está activo, pero el segundo no.

También podemos comprobar los logs del sistema en lb:

```
# tail /var/log/syslog
Sep 11 22:59:45 mail ldirectord[8543]: Added virtual server:
70.253.158.44:80
Sep 11 22:59:45 mail ldirectord[8543]: Added fallback server: 127.0.0.1:80
( x 70.253.158.44:80) (Weight set to 1)
Sep 11 22:59:45 mail ldirectord[8543]: Added real server: 70.253.158.41:80
( x 70.253.158.44:80) (Weight set to 1)
```



```
Sep 11 22:59:45 mail ldirectord[8543]: Deleted fallback server: 127.0.0.1:80
( x 70.253.158.44:80)
Sep 11 22:59:46 mail ldirectord[8543]: Added real server: 70.253.158.45:80
( x 70.253.158.44:80) (Weight set to 1)
```

De regreso a web1 y web2, comprobamos los logs de acceso de Apache. El director solicita director.html cada checkinterval segundos:

```
70.253.158.44 - - [11/Sep/2006:22:49:37 -0500] "GET /director.html
HTTP/1.1"
200 11 "-" "libwww-perl/5.803"
70.253.158.44 - - [11/Sep/2006:22:49:39 -0500] "GET /director.html
HTTP/1.1"
200 11 "-" "libwww-perl/5.803"
```

En su navegador, vaya a la URL del sitio virtual `http://70.253.158.42/which.html` y debería ver:

```
Yo soy web1
```

o:

```
Yo soy web2
```

Si el balanceador de carga no funciona o uno de los servidores Web está caído, siempre obtendrá la respuesta del mismo servidor Web.

Ahora, pare Apache en web1:

```
# /etc/init.d/apache stop
```

Actualice la página del navegador `http://70.253.158.42/which.html`.

Siempre debería obtener la respuesta:

```
Yo soy web2
```

Añadiendo HA a LB

El balanceador de carga representa un único punto de fallo. Si falla, lo servidores Web que dependen de él dejarán de ser accesibles. Para hacer el sistema más seguro, puede instalar un segundo balanceador de carga en configuración HA con el primero. Para instrucciones más detalladas (se usa el paquete Ultra Monkey que ya hemos instalado) puede consultar `http://www.howtoforge.com/high_availability_loadbalanced_apache_cluster`. No necesitará HA para los servidores Apache, puesto que ldirectord los sondea todos cada checkinterval segundos y ajusta todos los pesos, que tienen un efecto muy similar al hearbeat de HA.

Añadiendo otros servicios LB

Hemos usado servidores Web Apache como ejemplo para este capítulo porque es uno de los servicios que puede necesitar una granja de servidores. Otros servicios también podrían beneficiarse de LB/HA, por ejemplo MySQL, servidores de correo o servidores LDAP. Consulte http://www.howtoforge.com/loadbalanced_mysql_cluster_debian para un ejemplo de MySQL.

Escalabilidad sin LB y HA

Si ofreciera un magnífico servicio, ¿sería el servidor capaz de soportar grandes picos de actividad? Si no es así, su credibilidad se vería afectada y muchos visitantes no volverían. Pero debido a que LB y HA necesitan un importante esfuerzo e inversión en hardware, no está de más considerar otras posibilidades. Hay otras formas de conseguir un mayor rendimiento del equipo actual. Por ejemplo, puede desactivar los archivos .htaccess en su configuración Apache (AllowOverride None), o usar mod_expires para evitar llamadas de actualización de los archivos como imágenes. Hay muchos libros de Apache con trucos de optimización.

Una vez que se alcancen los límites del software del servidor Web, pueden considerarse otras alternativas. En muchos casos, servidores Web como lighttpd (http://www.lighttpd.net), Zeus (http://www.zeustech.net) y litespeed (http://litespeedtech.com) son más rápidos que Apache y consumen menos memoria.

También puede mejorar el rendimiento por otros métodos. Cachés de código, que incluyen aceleradores PHP como e-accelerator (http://eaccelerator.net) y APC (http://apc.communityconnect.com), guardan PHP como bytecode y así evitan tener que usar tiempo de cómputo en cada acceso a una página. Cachés de datos, como la caché de consultas de MySQl guardan los resultados de consultas idénticas. La replicación es una forma de LB. Memcached (http://danga.com/memcached) es una forma rápida de almacenar resultados de búsquedas. Squid (http://www.squidcache.org), cuando se usa como proxy inverso, es un caché de páginas que puede almacenar todo el contenido del servidor Web.

Cuando los servidores están separados en varias aplicaciones (por ejemplo MySQL ? PHP ? Apache), las mejoras son multiplicativas; por ejemplo "Getting Rich with PHP 5" (http://talks.php.net/show/oscon06) indica cómo conseguir que una aplicación PHP que soporta 17 llamadas por segundo pueda pasar a soportar 1.100 llamadas por segundo en una única máquina. Si ya está usando estás técnicas pero no solucionan sus problemas, definitivamente inténtelo con balanceo de carga, y si la estabilidad es crítica incorpore también HA.

Otras lecturas

Más detalles sobre el software usado en este capítulo están disponibles en las siguientes páginas Web:

- Linux Virtual Server Project (http://www.linuxvirtualserver.org).
- Ultra Monkey (http://www.ultramonkey.org).
- Heartbeat/The High-Availability Linux Project (http://linux-ha.orgS).

También tiene otra opción, y puede echar un vistazo a la Red Hat Cluster Suite (http://www.redhat.com/software/rha/cluster), un producto LB/HA comercial para Linux que usa LVS. El mismo software está disponible gratuitamente (pero sin soporte) en CentOS.

Capítulo 8

Servicios de red de área local



En este capítulo repasaremos algunas de las destrezas que un administrador de sistemas necesita para gestionar un equipo situado detrás del cortafuegos o la pasarela de una compañía, una organización o incluso una red local.

Algunos de nosotros preferimos referirnos a tecnologías de Internet en lugar de a redes de área local, puesto que creemos que no presenta ningún desafío. Pero cuando necesitamos configurar o arreglar algo en nuestro entorno de trabajo, la red de área local adquiere mucha importancia. Por ejemplo, piense que sucedería si el correo electrónico del directorio no funcionase.

Una red local se puede llevar la mayor parte del tiempo del administrador de sistemas si no sabe manejarla. Por lo que, si acaba de empezar en la administración de sistemas, necesitará aprender cómo instalar, configurar y mantener los servicios de una red local. Para la primera toma de contacto, eche un vistazo a la edición más reciente del libro "Linux Network Administrator's Guide Terry Dawson", (O'Reilly). Tanto si posee conocimientos básicos como usuario de Linux, como si nunca ha oído hablar sobre los temas de este capítulo, le aseguramos que son muy interesantes.

En este capítulo, exploraremos los sistemas de archivos distribuidos, como configurar los servicios de DHCP y de pasarela (incluyendo el routing entre la LAN e Internet), los sistemas de impresión corporativos y la gestión de usuario. Los servicios de correo electrónico local, también están bajo el paraguas de las LAN. Pero ya cubrimos esos aspectos en capítulos anteriores. Usaremos la distribución Fedora Core para este capítulo. Red Hat patrocina el proyecto Fedora y normalmente lo usa para probar la próxima distribución empresarial estable. Fedora no es la versión más estable de Red Hat, pero sí es razonablemente estable y robusta. Red Hat ofrece paquetes nativos de muchas herramientas para Fedora, dejando a Fedora en la vanguardia de las distribuciones libres de Linux, que pue-

den usarse para uso comercial. Si no le gusta el modelo de Red Hat, puede aplicar el material de este capítulo a otras distribuciones de Linux. Le sugerimos que profundice en este material: necesitará ponerlo en práctica en un entorno de trabajo, además no encontrará este material en otro sitio.

Sistemas de archivos distribuidos

Es difícil imaginar el tiempo en que los PC trabajaban de manera autónoma sin los beneficios del trabajo en red o la conexión a Internet. Pero los PC no fueron originalmente diseñados para trabajar en red. Puede que recuerde la época en que la gente transfería archivos entre equipos con disquetes, o usaban un conmutador para poder compartir una impresora. Eran tiempos difíciles.

Después de la introducción del PC, se tardó varios años en crear tecnología básica para el trabajo en red como los sistemas de archivos distribuidos. Estos sistemas de archivos distribuidos transformaron el paisaje de los negocios, debido a que permitían compartir lo que cada uno tenía en su escritorio. Ya no hacía falta rellenar manualmente un formulario para que el operador de un sistema mainframe le dejara hacer su trabajo. El trabajo en red se hizo más popular cuando un investigador de IBM, Barry Feigenbaum, convirtió un sistema de archivos DOS en uno distribuido. Sus esfuerzos ayudaron a crear el protocolo de aplicación *Server Message Block* (SMB). La era de los administradores de sistemas y de los ingenieros de red había comenzado.

Los sistemas de archivos distribuidos permiten a los usuarios abrir, leer y escribir archivos que estaban almacenados en otros ordenadores distintos al suyo. En algunos entornos, un único ordenador de altas prestaciones almacena los archivos, y los usuarios acceden a ellos a través de una LAN; el ordenador central puede incluso almacenar los directorios personales de los usuarios, por lo que todo el trabajo se almacena en él. En otros entornos, los usuarios almacenan archivos en sus PC, pero permiten a los otros usuarios acceder a estos archivos. Los dos entornos pueden combinarse. A esta práctica se la suele conocer como compartición de archivos, y a los directorios (o carpetas) a los que los usuarios pueden acceder desde máquinas remotas se les conoce como directorios compartidos. Los PC se convirtieron en los protagonistas de los negocios a finales de la década de 1980, y las redes de área local se dieron a conocer a medida que los PC evolucionaban y la gente se dio cuenta de que necesitaba compartir sus recursos.

Intente imaginar qué supuso la introducción de la LAN para un grupo de usuarios que nunca antes había trabajado en red. De repente, los compañeros de trabajo podían compartir documentos, imprimir en impresoras que no estaban en su despacho y responder a correos enviados por gente de su oficina, de su campus o de país. Todo esto supuso una verdadera revolución.

Hoy en día, muchos sitios almacenan sus usuarios en archivos críticos en servidores centrales, que controlan los permisos de acceso a los archivos. Hablaremos sobre gestión de usuarios más tarde en este mismo capítulo.

Introducción a Samba

La compartición de archivos y de impresoras con SMB evolucionó de la mano de Microsoft en el protocolo *Common Internet File System* (CIFS). CIFS ha sido publicado como un estándar, pero está poco documentado y contiene muchos comportamientos secretos que Microsoft sigue desarrollando. Sin embargo, un grupo intrépido de desarrolladores ha estado haciendo un trabajo de ingeniería inversa con el protocolo, y ha creado uno de los proyectos de software libre más populares y que puede usarse tanto desde sistemas Microsoft como otros sistemas: Samba. Samba está llegando a ser muy popular; puesto que se usa significativamente en Windows, en Linux e incluso en Mac OS X.

Como administrador de sistemas Linux, necesitará comprender Samba. Si desea profundizar en Samba (debería hacerlo), existen excelentes libros que tratan la materia, incluyendo guías de documentación en línea `http://samba.org`. Para usar una cita común, "Una descripción más a fondo de este tema sobrepasa el ámbito de este libro". Actualmente, no tenemos una razón para duplicar el excelente material que ya está disponible. Sin embargo, queremos describir Samba con suficiente detalle para hacer funcional su entorno. Afortunadamente, la mayoría de las distribuciones ofrecen interfaces gráficas muy sencillas que permiten administrar Samba, describiremos algunas de ellas aquí.

Algunas funciones principales de las redes CIFS (sobre todo la forma en que los sistemas se buscan unos a otros) tienen lugar en controladores de dominio: servidores que ofrecen archivos, impresoras y varias operaciones de control. Samba puede integrar máquinas Linux en redes de Microsoft como servidores de archivo e impresoras, controladores de dominio o miembros de un grupo de trabajo.

La última versión de Samba puede operar con el Active Directory de Microsoft. Samba combina LDAP con funciones como un servidor de autentificación robusta, sustituyendo tanto a los controladores de dominio de Microsoft NT como a los servidores Active Directory.

Samba también puede desempeñar un papel de compartidor de archivos en entornos más simples, donde los miembros de pequeñas oficinas o departamentos de una gran organización usan redes punto a punto. Los usuarios de escritorio pueden compartir sus impresoras y archivos con otros sin que éstos tengan que autentificarse. Si funciones sensibles como contabilidad financiera o almacenamiento de personal se gestionan en una máquina, pueden implantarse políti-

cas de seguridad que sirvan de escudo para otros usuarios sin comprometer la disponibilidad de los recursos de la red punto a punto.

Ahora, echaremos un vistazo a las redes Linux/Windows y veremos cómo puede configurar Samba para sus usuarios de escritorio.

Configurando la red

La figura 8.1 representa una red y puede verse desde un sistema Linux (la distribución Xandros, que es un entorno Linux de escritorio adecuado para entornos corporativos).

La vista de árbol de la parte izquierda de la pantalla muestra cuatro equipos llamados Athlon, Atlanta, Dallas y Dell. Dallas ofrece una impresora, junto con varios directorios, a los otros sistemas; Dell también aloja una impresora. Uno de los otros ordenadores ejecuta Windows XP, y los otros dos ejecutan Windows 98. Linux los agrupa a todos por igual. El sistema Linux hace lo mismo que un sistema Windows con la opción Mi entorno de red o Mis sitios de red.

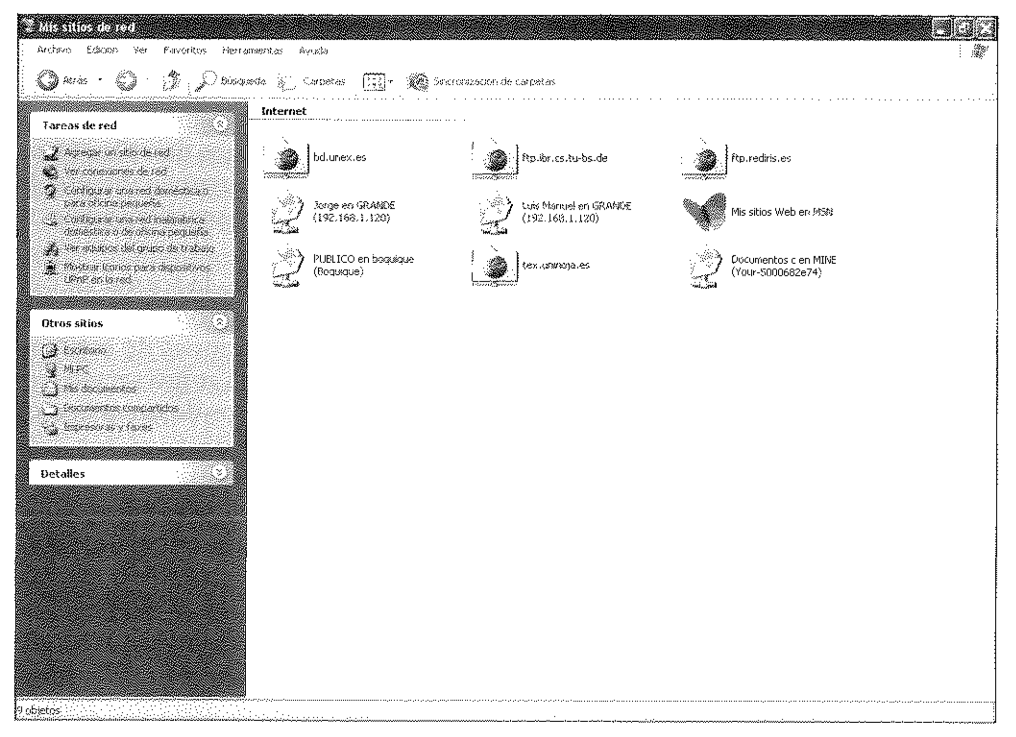


Figura 8.1. Los archivos y los directorios compartidos por un sistema Linux, tal y como se ve desde un PC con Windows.

En la parte derecha de la pantalla en la figura 8.1 destacan los directorios compartidos del nodo llamado Dallas, que es un sistema Windows XP. También puede ver el archivo de un procesador de textos llamado `xp_network_setup.sxw`, que se ha guardado en el formato nativo del OpenOffice Writer (Versión 1).

¿Fue difícil configurar la red? Aparte del cableado estándar, las conexiones Ethernet y la instalación del cortafuegos y del modem, el sistema se instala por sí solo. Hemos seguido los procedimientos estándar para Windows 98. Los sistemas usan DHCP para obtener sus direcciones IP, servidores DNS y la dirección hasta la pasarela. El router ofrece servicios DHCP y una dirección privada de Internet usando una red de clase C (desde 192.168.0.0 hasta 192.168.0.255). (Describiremos DHCP en la siguiente sección.)

Una vez que los sistemas Windows han establecido la configuración de la red y pueden conectarse a Internet, haga clic con el botón derecho del ratón sobre el icono **Mis sitios de red**, seleccione Propiedades y cambie las direcciones dinámicas por estáticas. Esto permite a los ordenadores actuar como servidores de impresión y ofrecer acceso compartido a Internet.

Configurar sistemas Windows XP es ligeramente más complicado, puesto que XP y Windows 98 son incompatibles entre sí. Para asegurarse de que se entienden, active el *Simple File Sharing* accediendo al Panel de Control de Windows XP y ejecutando el Asistente de Configuración de Red. Este asistente nos pregunta si queremos activar la compartición con otros equipos, refiriéndose a ordenadores con Windows 98. Al responder que sí se permite crear un disquete que podríamos usar para instalar los protocolos de Windows XP en ordenadores con Windows 98. Este proceso actualiza los sistemas más viejos a protocolos más nuevos, ofreciendo la posibilidad de que XP y Windows 98 puedan comunicarse. (El programa ofrecido por Microsoft se llama netsetup.exe.)

Luego se instala la distribución Xandros y se activa el Windows Networking, tal y como se muestra la figura 8.2.

Fíjese en que hemos sido capaces de configurar Windows Networking a través del cuadro de diálogo. El escritorio Linux nos ha permitido activar la compartición de archivos y de impresoras, dar un nombre al equipo, definir el grupo de trabajo y activar el nivel de seguridad compartida, que permite a los nodos usar la funcionalidad CIFS.

Ubuntu también le da la opción de configurar el *Network File System* (NFS), un sistema de archivo Unix-a-Unix a muy popular que es incompatible con CIFS. El cuadro de diálogo de la figura 8.4 le permite escoger entre ambos sistemas; puede usar Samba para operar con Windows y Mac OS X, mientras usa NFS para operar con otros sistemas Unix/Linux. Los servicios de compartición no se instalan por defecto con Ubuntu, pero si selecciona Carpetas Compartidas (en el menú de administración de Ubuntu 6.10), Ubuntu descargará los archivos necesarios; ahora ya está listo para ser miembro del dominio o del grupo de trabajo.

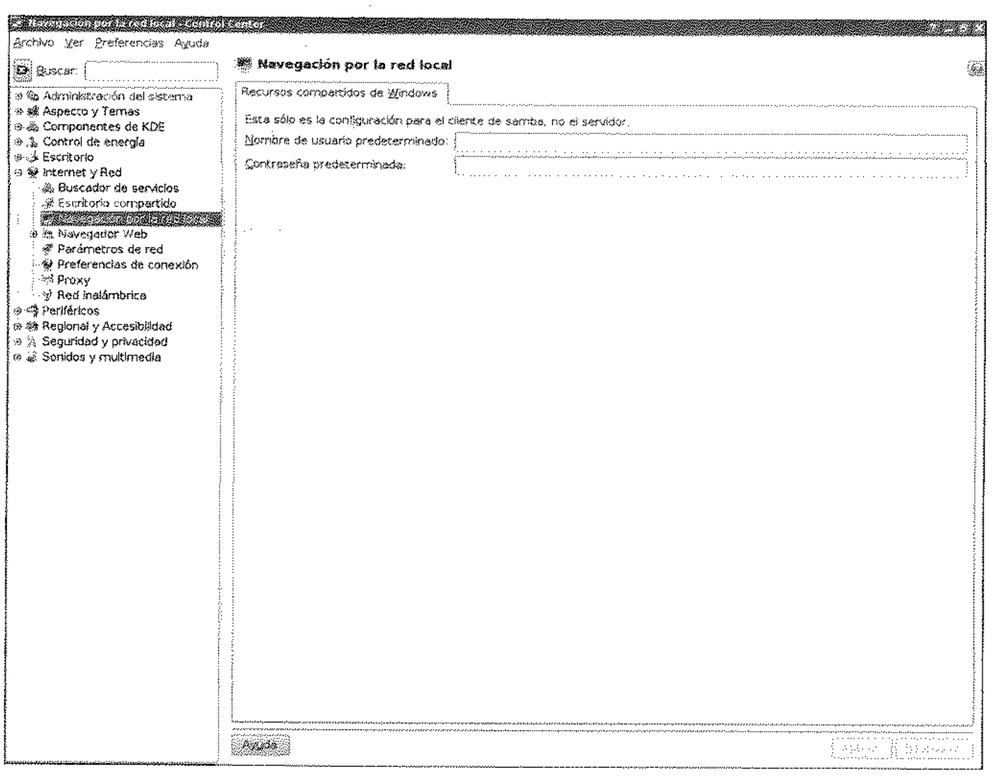


Figura 8.2. ConfigurandoWindows Networking.

Figura 8.3. Configurando los recursos compartidos de Ubuntu en un entorno Windows.

Profundizaremos un poco más en aspectos de Samba más adelante en este mismo capítulo.

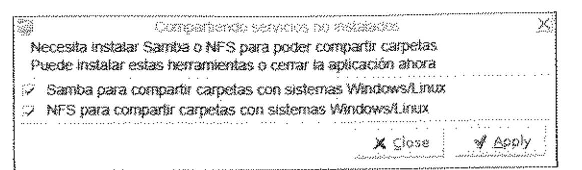


Figura 8.4. Pantalla de configuración de los servicios de compartición de archivos en Ubuntu.

DHCP

Los servicios de *Dynamic Host Configuration Protocol* (DHCP) pueden ayudarle a solucionar un gran número de problemas asociados con entornos de área local, incluyendo problemas de asignación de direcciones IP y otros aspectos de administración. Es difícil imaginar una red sin DHCP.

Veamos algunos aspectos que debe conocer y prestemos atención a cómo pueden ayudarnos:

- Los PC y las estaciones de trabajo necesitan una dirección IP única, información DNS y conocer la ubicación de las pasarelas.
- La asignación manual de direcciones IP provoca excesivo trabajo.
- La duplicación accidental de una dirección IP crea conflictos en la red.
- La resolución de problemas de red (como la duplicación de direcciones) y los cambios de ubicación provocan trabajo innecesario.
- Los cambios de personal significan que alguien tendrá que comprobar cada ordenador y configurar una nueva base de datos de asignaciones de IP.
- El movimiento frecuente de los usuarios móviles necesita que se reconfigure la red en los equipos portátiles.

DHCP soluciona estos problemas asignando direcciones IP a medida que cada sistema de la LAN arranca. El servidor DHCP asegura que todas las direcciones IP son únicas. El servicio necesita una pequeña intervención humana relativa a la asignación y al mantenimiento de las direcciones IP. Los administradores pueden escribir archivos de configuración y dejar el resto del trabajo al servidor DHCP (dhcpd). Este servidor gestiona el conjunto de direcciones IP, liberando al administrador de red de esta tarea.

Instalando DHCP

Para comenzar con DHCP, primero necesita instalar el servidor DHCP. Puesto que este capítulo se basa en Fedora, puede instalar el paquete RPM con Yum o

con el gestor de paquetes /usr/bin/gnome-app-install; la versión actual del paquete es dhcp-3.0.3-28.i386. (Los usuarios Debian pueden instalar el paquete dhcp3-server y editar el archivo de configuración /etc/dhcp3/dhcpd.conf). El software fue creado por *Internet Systems Consortium*.

Una vez lo haya instalado, configure DHCP en /etc/dhcpd.conf. Como primer paso, copie el archivo /usr/share/doc/dhcp/dhcpd.conf.sample a /etc/dhcpd.conf. Luego, edite el archivo para que se ajuste a su red. El siguiente ejemplo es típico. La sintaxis usa almohadillas (#) para los comentarios:

```
ddns-update-style interim;
ignore client-updates;

subnet 192.168.1.0 netmask 255.255.255.0 {

# -- default gateway
   option routers 192.168.1.1;
   option subnet-mask 255.255.255.0;

# -- option nis-domain "domain.org";
# -- option domain-name "domain.org";
option domain-name-servers 192.168.1.1;

# -- option time-offset -18000; # Eastern Standard Time
# option ntp-servers 192.168.1.1;
# option netbios-name-servers 192.168.1.1;
# -- Selects point-to-point node (default is hybrid). Don't change this
# -- unless you understand Netbios very well
# option netbios-node-type 2;
# -- range dynamic-bootp 192.168.0.128 192.168.0.254;
   default-lease-time 21600;
   max-lease-time 43200;

# we want the nameserver to appear at a fixed address
host ns {
   next-server server1.centralsoft.org;
   hardware ethernet 00:16:3E:63:C7:76;
   fixed-address 70.253.158.42;
   }
}
```

Hemos configurado unos cuantos elementos de nuestro archivo de configuración después de haberlo copiado al directorio /etc:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
   option routers 192.168.1.1;
   option domain-name-servers 192.168.1.1;
   option subnet-mask 255.255.255.0;
   default-lease-time 21600;
   max-lease-time 43200;
```


La primera línea define el rango del conjunto de direcciones IP disponibles para los usuarios de la subred de la LAN. En este caso, hemos usado la red privada de Clase C 192.168.1.0 que ofrece 254 nodos (desde el 192.168.1.1 hasta el 192.168.1.254). Esta máscara de red debe coincidir con la máscara de red usada para definir su LAN.

Hemos especificado la dirección de la pasarela en la segunda línea (option router) y el servidor de nombres en la tercera línea (option domain-name-servers). La dirección IP es la misma en las dos líneas, puesto que es una práctica común hacerlo así.

Un servidor simple con dos tarjetas de red a menudo actúa como pasarela de una red de área local. Una tarjeta, representada por un nombre de dispositivo como eth0, tiene una dirección en Internet, mientras que la otra tarjeta (supongamos que es eth1) tiene una dirección en una red privada. Cuando el reenvío de paquetes y el cortafuegos iptables están activados, cualquier servidor Linux puede actuar como pasarela/cortafuegos. En este caso, también se puede activar BIND en modo caché para que funcioné como el servidor DNS de la red.

Las últimas dos líneas especifican la cantidad de tiempo que un cliente puede mantener la dirección, se mide en segundos.

En nuestro archivo de configuración DHCP, también hemos añadido una cláusula para especificar la dirección estática de un servidor DNS corporativo.

```
# we want the nameserver to appear at a fixed address
host ns {
   next-server server1.centralsoft.org;
   hardware ethernet 00:16:3E:63:C7:76;
   fixed-address 70.253.158.42;}
```

En la próxima sección describiremos cómo usar dhcpd para asignar direcciones IP estáticas basadas en la dirección MAC de la tarjeta de red del cliente. Pero antes de hacer eso, veamos una versión sencilla de /etc/dhcpd.conf:

```
ddns-update-style interim;

default-lease-time            600;
max-lease-time                7200;

subnet 192.168.1.0 netmask 255.255.255.0 {
   option routers 192.168.1.1;
   option subnet-mask 255.255.255.0;
   option domain-name-servers server.centralsoft.org,
      server2.centralsoft.org;
   range 192.168.1.2 192.168.1.254;
}
```

Nota: Para servidores DHCP sencillos, el mantenimiento será más sencillo si omite los comentarios y hace que el archivo de configuración sea pequeño.

Iniciando el servicio DHCP

Algunos servicios DHCP necesitan un archivo dhcpd.leases. Use el comando touch para crear un archivo vacío en el mismo directorio que el archivo `dhcpd.conf`:

```
# touch /var/lib/dhcp/dhcpd.leases
```

Antes de iniciar el servidor DHCP, compruebe que la configuración es correcta. También sería recomendable configurar que el servidor se inicie al arrancar. Para iniciar el servidor, introduzca:

```
[root@host2 ~]# service dhcpd start
Starting dhcpd:              [ OK ]
[root@host2 ~]#
```

También puede comprobar si el proceso DHCP se está ejecutando con el siguiente comando (si el servicio se está ejecutando, se mostrará una línea con las estadísticas del proceso):

```
# ps aux | grep dhcpd
root  9028  0.0   0.0   2552  636   Ss     09:40     0:00 /usr/sbin/dhcpd
```

Use el comando chkconfig para conseguir que DHCP se inicie al arrancar:

```
# chkconfig dhcpd on
# chkconfig --list
....from the list:
dhcpd     0:off     1:off     2:on     3:on     4:on     5:on     6:off
```

Al igual que sucede con los otros servicios de Linux, tendrá que reiniciar el demonio DHCP siempre que haga cambios en los archivos de configuración. Puede establecer otras opciones en el archivo `dhcpd.conf` de manera global o para el cliente de un equipo o para una subred. Esto quiere decir que puede establecer una configuración por defecto para su red, y luego sobrescribirla para un cierto grupo de máquinas e incluso para máquinas individuales. Aquí hay un ejemplo de una sección de configuración global en la parte de arriba del archivo `dhcpd.conf`:

```
option domain name "host2.centralsoft.org";
```

Ofreciendo direcciones IP estáticas

Las estaciones de trabajo suelen funcionar bien con direcciones dinámicas (es decir, direcciones que pueden cambiar periódicamente o después de reiniciar), pero los servidores normalmente necesitan direcciones estáticas, por lo que sus direcciones no cambian mientras están atendiendo a un cliente. Por ello, DHCP

permite especificar direcciones IP estáticas para sistemas particulares en `dhcp.conf`. Vamos a hacerlo. Primero, configure la subred, la dirección de broadcast y los routers:

```
subnet 192.168.1.0 netmask 255.255.255.0
option broadcast-address 192.168.1.255;
option routers 192.168.1.1;
```

Luego, añada una sección host por cada máquina de la red. Para hacer esto, necesitará saber la dirección hardware (a menudo llamada dirección MAC) de cada tarjeta de red, que podrá averiguar usando el comando ifconfig en cada equipo. Aquí mostramos un ejemplo de una sección host:

```
# ethernet MAC address as follows (Host's name is "laser-printer"):

host laser-printer {
   hardware ethernet 08:00:2b:4c:59:23;
   fixed-address 192.168.1.10;
}

host1.centralsoft.com {
   hardware ethernet 01:0:c0:2d:8c:33;
   fixed-address 192.168.1.5;
}
```

Cree una cláusula de configuración como esta para cada servidor que necesite una dirección IP estática y añádala al archivo de configuración.

Asignando direcciones IPv6 con radvd

A finales de 1995, Steve Deering y Robert Hinden se dieron cuenta de la necesidad de un nuevo protocolo de direccionamiento para Internet. Su primera especificación de IPv6 apareció en 1995, en el IETF *Request Ror Comments* (RFC) 1883; la segunda apareció en 1998, en el RFC 2460. Deering y Hinden argumentaban lo que mucha gente ya sabía: que el espacio de direcciones de IPv4 de 32 bits iba a limitar el crecimiento explosivo de Internet.

Algunos administradores de sistemas se han dado cuenta de que IPv6 y sus métodos para asignar direcciones IP han empezado a ganar popularidad, Aunque mucha gente duda de IPv6, argumentando que es innecesario o que en nunca se va a imponer en la práctica real, hay muchas aplicaciones y muchos entornos que están girando en esta dirección.

Nota: Una descripción exhaustiva está, de nuevo, fuera del alcance de este libro; para más información sobre el protocolo IPv6 y su demonio, así como para saber cómo obtener direcciones IPv6 públicas, tendrá que buscar en otros sitios.

Las direcciones IPv6 a menudo incluyen la dirección hardware de la tarjeta de red. Esta propiedad permite a los usuarios de IPv6 obtener la dirección IP estática sin necesitar ninguna configuración en la parte del servidor que soporte estas direcciones. La asignación automática de direcciones IPv6 puede hacerse con la ayuda del demonio router-advertising radvd. Los usuarios de Fedora pueden instalar el paquete radvd-0.9.1 desde los repositorios Yum. Los usuarios Debian pueden instalar el paquete radvd y leer el archivo `/usr/share/doc/radvd/README.Debian`.

radvd escucha las peticiones que se hacen al router y envía mensajes tal y como se describe en el RFC 2461, "Neighbor Discovery for IP Version 6 (IPv6)". Los equipos pueden configurar automáticamente sus direcciones y escoger sus routers por defecto basándose en estos mensajes.

radvd soporta un protocolo sencillo. Su instalación también es sencilla. Un ejemplo de configuración para el archivo `/etc/radvd.conf` sería este:

```
interface eth0
{
    AdvSendAdvert on;
    prefix 0:70:1f0Q:96::/64
    {
    };
};
```

Si quiere usar radvd, necesitará cambiar el prefijo por uno de su red y configurar el servicio. También tendrá que configurar el DNS en las estaciones de trabajo clientes de forma separada.

Puede encontrar variada información sobre el proyecto radvd en la página `http://www.litech.org/radvd`.

Servicios de pasarela

Linux ofrece facilidades a los usuarios para que naveguen por Internet sin exponer sus direcciones IP individuales al público. La configuración típica oculta a una organización del público usando Linux como router. En la parte privada del router, las actividades locales no pueden ser detectadas por nadie de la parte pública.

La gente a veces se refiere a una pasarela como a un *bastion host*. Lo correcto es pensar que es una entidad de red que ofrece una única entrada y una única salida hacia Internet. El *bastion host* ayuda a evitar que se comprometa la seguridad de una red ofreciendo una barrera entre las áreas pública y privada. Nos referiremos a los servicios que ofrece como servicios de pasarela.

Los administradores de sistemas Linux implantan servicios de pasarela usando una combinación de reenvío de paquetes y reglas de cortafuegos conocidas

como iptables. También hay disponibles otros servicios de pasarela, como el enmascaramiento o el *Network Address Translation* (NAT).

En organizaciones pequeñas y en redes locales, una pasarela puede implantarse en un único servidor incluyendo seguridad básica, un cortafuegos, DHCP, DNS caché y servicios de correo. En organizaciones más grandes, tales servicios están normalmente repartidos por varios servidores, con una zona desmilitarizada (DMZ) actuando como pasarela.

El papel de una DMZ

En seguridad informática, se conoce con el término zona desmilitarizada a una red perimetral, que es una subred o una red situada entre la red interna e Internet. Por ejemplo, su red privada debería usar una red privada del tipo 192.168.1.0, la DMZ 10.0.0.0 y el bloque público de Internet 70.253.158.0.

Las DMZ se usan para contener servidores que necesitan ser accesibles desde fuera, tales como servidores de correo, servidores Web o servidores de DNS. Las conexiones desde Internet a la DMZ suelen controlarse usando la función *Port Address Translation* (PAT).

Una DMZ se suele situar en el medio de dos pasarelas o cortafuegos y conecta a ambos, con una tarjeta de red conectada a la red interna y otra conectada a Internet. Una DMZ puede evitar fallos de configuración accidentales que podrían permitir el acceso desde Internet a la red interna. A esto se le llama cortafuegos de subred apantallada.

Para nuestros propósitos, será suficiente con limitar la configuración de la pasarela para el reenvío de paquetes; no gastaremos tiempo en la DMZ, que necesita mucho equipamiento y esfuerzo. Para levantar una pasarela necesitará:

- Un ordenador dedicado que actúe como pasarela.
- Una conexión a Internet y dos tarjetas de red.
- Un pequeño conmutador para que las máquinas clientes se conecten a la pasarela.
- iptables instalado.

Supondremos que eth0 es su conexión de Internet y eth1 es su pasarela interna en esta configuración. Edite el archivo de configuración para eth0, que está en `/etc/sysconfig/networking/devices/ifcfg-eth0` para incluir las siguientes líneas:

```
ONBOOT=yes
USERCTL=no
IPV6INIT=no
PEERDNS=yes
GATEWAY=70.253.158.46
TYPE=Ethernet
```

```
DEVICE=eth0
HWADDR=00:04:61:43:75:ee
BOOTPROTO=none
NETMASK=255.255.255.248
IPADDR=70.253.158.43
```

Asimismo, la configuración para eth1 debería ser:

```
ONBOOT=yes
USERCTL=no
IPV6INIT=no
PEERDNS=yes
TYPE=Ethernet
DEVICE=eth1
HWADDR=00:13:46:e6:e5:83
BOOTPROTO=none
NETMASK=255.255.255.0
IPADDR=192.168.1.1
```

La información de estos parámetros de configuración puede encontrarse en el archivo `sysconfig.txt`, que encontrará en `/usr/share/doc/initscripts-7.93.7`. Con las tarjetas de red configuradas, necesitará asegurarse de que ha instalado iptables. Debería ver el siguiente resultado:

```
[root@host2 devices]# rpm -q iptables
iptables-1.3.5-1.2
[root@host2 devices]#
```

Si no tiene iptables instalado, instálelo ahora y cargue los módulos.

Nota: Fedora 5 instalará iptables usando la aplicación Instalar/Desinstalar Software, ubicado en el directorio por encima del menú Aplicaciones en el panel GNOME. También carga módulos del kernel como parte del proceso de instalación.

Luego ejecute:

```
# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
# service iptables save
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Ahora edite `/etc/sysctl.conf`, cambiando `net.ipv4.ip_forward = 0` por 1 para mantener esto activo tras reiniciar. Puede hacer que el sistema relea `/etc/sysctl.conf` tecleando:

```
# sysctl -p
```

Finalmente, si tiene una pequeña organización, puede añadir DHCP al servidor usando una versión sencilla de `dhcpd.conf`:


```
ddns-update-style interim;

default-lease-time            600;
max-lease-time                7200;
subnet 192.168.1.0 netmask 255.255.255.0 {
option routers 192.168.1.1;
option subnet-mask 255.255.255.0;
option domain-name-servers server1.centralsoft.org,
   server2.centralsoft.org;
range 192.168.1.2 192.168.100.254;
}
```

Otra aproximación a los servicios de pasarela

Esta sección cubre el uso de productos que son combinación de pasarela y cortafuego y que ofrecen conjuntos de múltiples funcionalidades. Existen varios paquetes gratuitos como Firestarter, IPCop, Netfilter y Shorewall. Verá como en la literatura Linux también se menciona Smoothwall y ClarkConnect, pero son productos comerciales que instalan una distribución Linux entera, no aplicaciones independientes.

Para su uso correcto en este capítulo, hemos elegido Firestarter. No obstante, puede que quiera echarle un vistazo a Shorewall, una utilidad de configuración para Netfilter (una herramienta de línea de comandos). Puede descargar Firestarter desde los repositorios de Fedora. Nuestra instalación necesita el siguiente paquete:

```
[root@host2 ~]# rpm -q firestarter
firestarter-1.0.3-11.fc5
[root@host2 ~]#
```

El asistente para Firestarter (figura 8.5) se lanza cuando un administrador inicia el programa por primera vez. Puede relanzar el asistente desde el menú del cortafuegos en la interfaz principal, así como cambiar las opciones desde la opción Preferencias.

Después de la pantalla de bienvenida inicial, habrá una serie de pantallas de configuración, comenzando por la pantalla de configuración del dispositivo de red (figura 8.6), que puede configurar tarjetas de red duales.

La función principal de Firestarter es la de compartir conexiones. Sin embargo, desde que usa NAT, funciona como una pasarela, por lo que los PC de una LAN es como si fueran una única máquina con una única dirección IP para Internet. Esto es evidente, por ejemplo, en la pantalla de preferencias mostrada en la figura 8.7. Fíjese en que la descripción del primer dispositivo se refiere a un "Dispositivo de red conectado a Internet" y la segunda descripción se refiere a una "Red local conectada a un dispositivo".

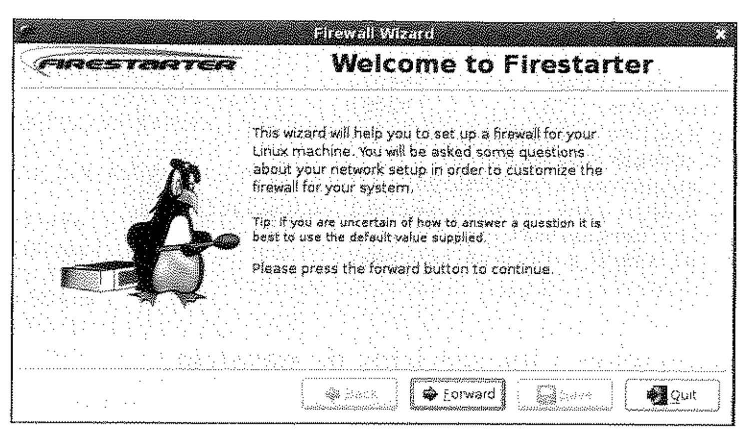


Figura 8.5. Asistente para el cortafuegos Firestarter.

Figura 8.6. Pantalla de configuración del dispositivo de red.

También se puede ver en la parte de abajo de la figura 8.7 que Firestarter permite al administrador usar una configuración DHCP existente para crear una nueva. Aquí mostramos el archivo `dhcp.conf` de Firestarter:

```
# DHCP configuration generated by Firestarter
ddns-update-style interim;
ignore client-updates;

subnet 192.168.1.0 netmask 255.255.255.0 {
   option routers 192.168.1.1;
   option subnet-mask 255.255.255.0;
   option domain-name-servers 70.253.158.42, 70.253.158.45, 151.164.1.8;
   option ip-forwarding off;
   range dynamic-bootp 192.168.1.10 192.168.1.254;
```



```
      default-lease-time 21600;
      max-lease-time 43200;
   }
```

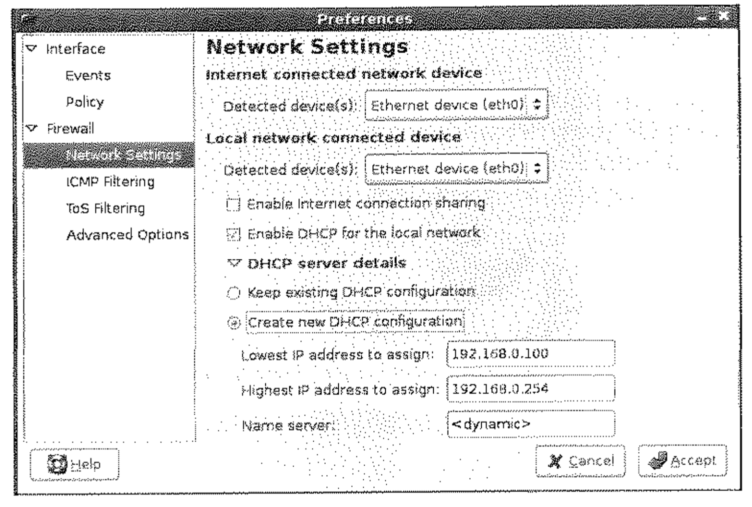


Figura 8.7. Pantalla de preferencias para Firestarter.

El archivo `resolv.conf` de la pasarela contiene la configuración DHCP de las máquinas clientes, por lo que Firestarter lee el archivo y coloca las direcciones del servidor DNS en `dhcpd.conf`.

La interfaz principal de Firestarter ofrece una vista del estado de la pasarela y las conexiones a los equipos DHCP. También ofrece un resumen de eventos y de actividad, tal y como se muestra en la figura 8.8.

En la figura 8.9, puede ver una vista de la segunda pestaña de la interfaz principal. En esta vista, puede ver las conexiones bloqueadas.

El panel Eventos ofrece un log de intentos de saltarse el cortafuegos. Puede ser útil para saber que hay intrusos que intentan entrar en sus sistemas. Si los intentos son reiterados, añada sus direcciones IP al archivo `/etc/host.deny`. Si alguien intenta entrar mediante ssh por el puerto 22 usando un ataque de diccionario, simplemente puede cerrar el puerto con Firestarter.

El icono de Firestarter se vuelve rojo cuando detecta algún riesgo potencial. Fíjese en el mensaje de la figura 8.10: "Detectado intento de conexión..." Merece la pena investigarlo.

La tercera pestaña de la interfaz principal permite definir las políticas para los servicios que se deben permitir y los que no. Por ejemplo, permitimos conexiones SSH, por lo que establecemos la política de permitir SSH en el puerto 22.

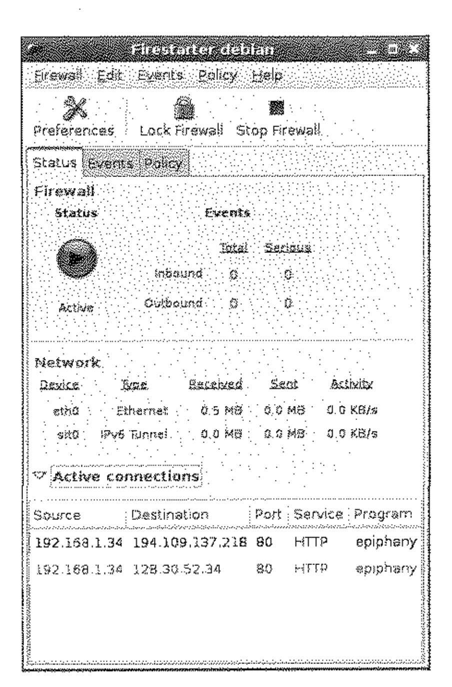


Figura 8.8. Interfaz principal de Firestarter.

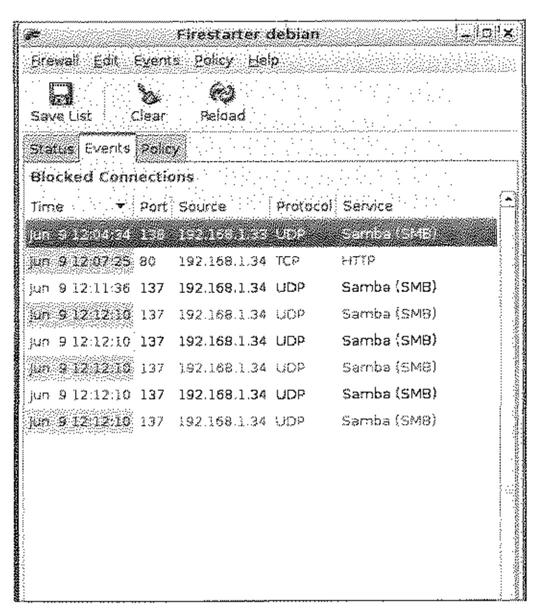


Figura 8.9. Panel de eventos de Firestarter.

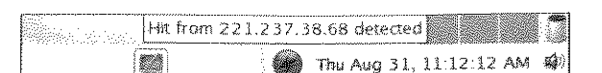


Figura 8.10. Icono mostrando un intento de intrusión.

Firestarter usa un asistente para configurar las políticas de pasarela. Puede hacerse una idea de cómo funciona viendo la figura 8.11.

La figura 8.11 muestra una ventana llamada "Añadir una nueva regla de entrada". Esta regla aparece después de que seleccione Añadir Regla en la pestaña Política. En esta ventana, puede ver una selección de opciones que pueda usar para permitir servicios dentro de la red. También existe una pantalla similar para ofrecer servicios externos a los usuarios.

Verá que Firestarter es una aplicación muy fácil de configurar. La comunidad que soporta el proyecto ha hecho un gran trabajo; ha documentado los procedimientos suficientemente y ha creado una guía del usuario que puede encontrar en `http://fs-security.com/docs.php`.

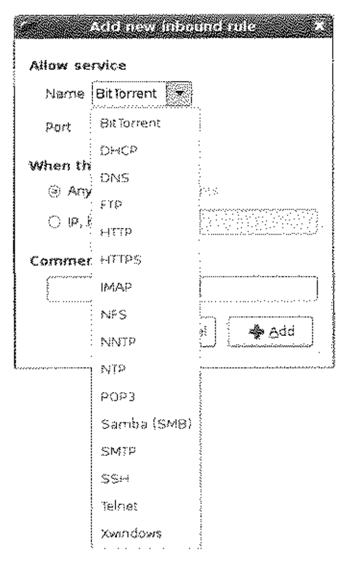


Figura 8.11. Configuración de políticas para Firestarter.

Nota: Llegados a este punto, puede preguntarse por qué hemos incluido una aplicación dependiente del entorno gráfico GNOME. Recuerde que cuando escogimos Fedora como distribución para el trabajo en red local, lo hicimos debido a su gran conjunto de herramientas. Añadir Firestarter encaja dentro de nuestra filosofía, téngase en cuenta la posibilidad de usar la interfaz de línea de comandos.

Servicios de impresión

Como administrador de sistemas Linux, debe saber que las impresoras pueden darle más de un quebradero de cabeza. A menudo encontrará que el hardware, el software y el sistema operativo son incompatibles. Debido a la amplia variedad de sistemas y de métodos para configurar las impresoras, esta área de administración puede conseguir ponerle de mal humor durante unos meses, o al menos hasta que se haga con la situación.

Vamos a empezar por el hardware. La mayoría de los administradores descubrirán cuatro tipos de hardware para impresoras en red. En las redes existentes, puede encontrar algunos de ellos ya configurados:

- Impresoras asignadas a equipos PC.
- PC dedicados como servidores de impresión.
- Impresoras de red con tarjetas Ethernet.
- Dispositivos servidores de impresión conectando directamente impresoras a la LAN.

En oficinas de tamaño medio, probablemente verá varias de estas soluciones muy a menudo. La flexibilidad ofrecida por los sistemas de escritorio modernos a menudo causa problemas.

Supongamos que uno de sus usuarios, Sally Jean, compra una impresora de inyección de tinta, solicita la compra como un gasto menor y la consigue. Luego, la conecta directamente a su PC. Billy Bob, que se sienta en la mesa de al lado, le pregunta que si puede usar su impresora, por lo que ella hace clic con el botón derecho del ratón sobre el icono de la impresora y selecciona Compartir. Billy Bob se intenta conectar a la impresora de Sally, pero no lo consigue. ¿Por qué? Él no tiene el driver instalado.

Así es que, estos dos usuarios llaman al administrador de sistemas (es decir, a usted) para que les solucione el problema. Usted instala el driver en el PC de Billy Bob, y de repente, como por arte de magia, funciona. Más tarde, Sally Jean le llama y se queja de que su PC necesita más memoria y un procesador más rápido. ¿Por qué? Hay diez personas usando su impresora porque la ha compartido, y eso hace que todo vaya más lento.

Cuando usted comprueba la situación, puede ver que hay una impresora láser que soporta grandes cargas de trabajo en la esquina y que nadie la utiliza. ¿Por qué nadie la utiliza? Sólo hay que investigar un poco para darse cuenta de que nadie la ha agregado al controlador de dominio.

Lo que esta hipotética anécdota muestra es que usted, como administrador de sistemas, necesita preparar una estrategia para gestionar la infraestructura de las impresoras. Esta sección del capítulo le ofrecerá una visión de alto nivel, así como suficiente información práctica para comenzar. Puede empezar el proceso haciendo un inventario del hardware y tomar decisiones respecto al software y a los sistemas operativos.

Debido a que hay muchos tipos de impresoras y combinación de dispositivos, sistemas operativos y software, tendrá que aprender la mayor parte de lo relacionado con la configuración de impresoras mientras practica. La mejor manera de aprender sobre impresión en red es intentar desarrollar una estrategia sobre una infraestructura propia. Eso reduce la cantidad de información que necesita procesar.

Consideraciones sobre el software de impresión

Linux y Windows empezaron a trabajar con modelos de impresión completamente diferentes. Afortunadamente, se han hecho progresos de cara a que puedan cooperar. Pero hasta que no configure las impresoras de su red, no se dará cuenta de que no son totalmente compatibles.

Originalmente, Linux solía usar el estándar Unix para impresión conocido como el Line Printer Daemon (LPD); más tarde, se añadió un demonio actualizado llamado LPRng. Las distribuciones Linux también solían usar herramientas LPD para imprimir e interoperar con variantes de Unix. Los distribuidores Linux continúan incluyendo LPD y sus herramientas, pero también han añadido soporte para un nuevo sistema conocido como *Common Unix Printing System* (CUPS). Al contrario que LPD, CUPS es compatible con Windows y con Mac OS. CUPS y LPD usan diferentes protocolos de impresión. Así como LDP no puede consultar aspectos básicos de un trabajo de impresión, CUPS sí puede. CUPS también trabaja directamente en redes heterogéneas y puede adaptarse a Samba si fuera necesario. No todas las distribuciones Linux activan la interfaz, pero Red Hat incluye Fedora por defecto.

Como administrador de sistemas, tendrá que familiarizarse con las herramientas administrativas de CUPS. En Fedora, simplemente teclee `http://localhost:631` en un navegador y podrá ver la interfaz de gestión presentada en la figura 8.12.

La interfaz es auto-explicativa, por lo que le dejaremos la exploración a usted. Si no tiene familiaridad con CUPS, eche un vistazo a la interfaz de gestión o vaya al sitio Web `http://www.cups.org/book/index.php` y lea el libro.

Impresión en plataforma cruzada

Ahora consideremos algunos dilemas a la hora de imprimir con los que tendrá que enfrentarse cada día en entornos empresariales. Casi seguramente se encontrará con situaciones donde necesitará compartir impresoras Linux con

máquinas Windows. (De hecho, probablemente necesite usar Linux como servidor de impresión en una red Windows para ahorrarse el precio de las licencias.) También puede necesitar compartir impresoras Windows en máquinas Linux ¿Cómo hacerlo?, echemos un vistazo a cómo los usuarios de Windows acceden a las impresoras Linux. Básicamente, necesitará definir un grupo de trabajo o un dominio Samba, y necesitará instalar CUPS en su PC Linux. También necesitará configurar CUPS para Samba, que puede hacerse usando el siguiente comando:

```
# ln -s 'which smbspool' /usr/lib/cups/backend/smb
```

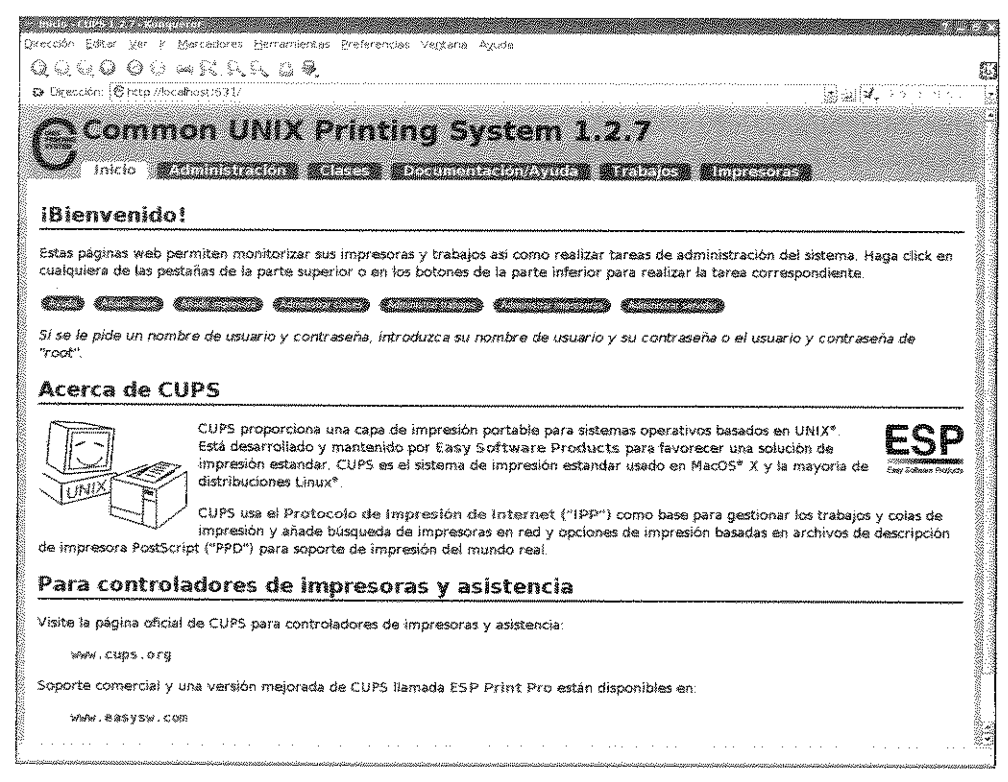


Figura 8.12. Interfaz de configuración de CUPS.

Edite `/etc/samba/smb.conf` para compartir una impresora en un servidor Samba. En una situación de la vida real es necesario restringir el acceso a ciertos sistemas o usuarios para cada impresora, pero en el siguiente ejemplo el PC Linux compartirá todas sus impresoras con cualquier sistema de la red en el que se haya configurado Samba:


```
[printers]
   comment = All Printers
   printing = cups
   printcap name = cups
```

Su PC Windows ahora puede acceder a las impresoras de la red. Probablemente necesitará los drivers de impresión de Windows, incluso los drivers que entraron con su impresora.

En el siguiente escenario, necesitará permitir a los usuarios Linux que usen las impresoras conectadas a los servidores Windows. De nuevo, necesitará CUPS y Samba para hacer esto. En los PC con Windows, comparta las impresoras como lo haría normalmente: en Windows NT, 2000 y/o XP active la cuenta de invitado y otorgue permisos a cualquiera que quiera acceder a la impresora compartida. Luego instale CUPS en el servidor Samba y configúrelo como se describió previamente.

Ahora instale las impresoras de Windows que quiera que estén disponibles en el servidor Samba con CUPS, usando para ello la interfaz Web.

Necesitará autentificarse como root. En algunos sistemas Linux, es necesario definir al root como el administrador del sistema CUPS. Puede hacer eso con el comando adduser:

```
~$ su
Password:
# adduser cupsys shadow
Adding user 'cupsys' to group 'shadow'...
Done.
# /etc/init.d/cupsys restart
Restarting Common Unix Printing System: cupsd                [ ok ]
#
```

Luego ya puede autentificarse como root.

Haga clic en Añadir impresora y luego introduzca el nombre de la impresora desde el sistema Windows. Vamos a usar BrotherHL1440 (véase figura 8.13). Luego, introduzca la ubicación y la descripción. Cuando llegue a la ventana del dispositivo, haga clic en la lista desplegable y seleccione Impresora en Windows vía Samba.

En la siguiente ventana, Dispositivo URI para, introduzca la URI del dispositivo. BrotherHL1440-2 está conectada a Philadelphia en Windows 2003, por lo que debe introducir el nombre de usuario "guest" y el nombre del equipo:

```
smb://guest@philadelphia/brotherhl1440-2
```

Llegados a este punto, tiene que seleccionar el driver de la impresora. También debería imprimir una página de prueba. En su cliente Linux, abra la interfaz CUPS y podrá ver la impresora. Los clientes Linux de la LAN ya pueden usar esta impresora.

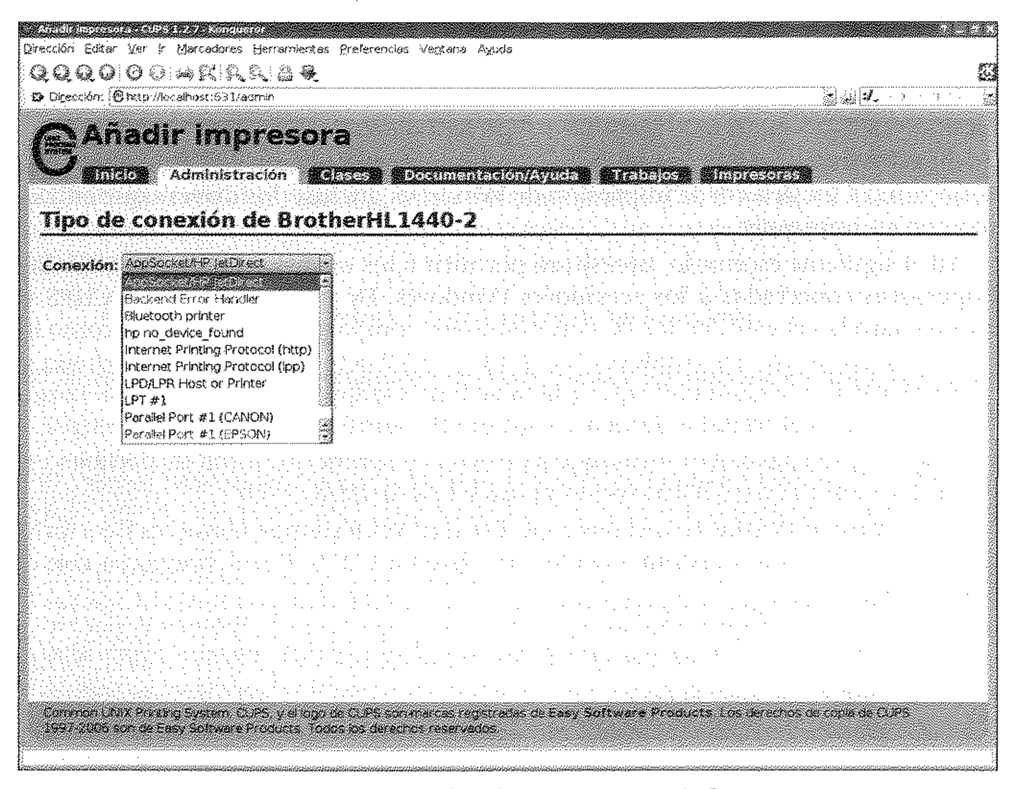


Figura 8.13. Añadiendo impresoras Windows.

Controlando las colas de impresión desde la línea de comandos

Puede hacer ssh a un servidor Linux de impresión remoto y usar los comandos CUPS para controlar las colas de impresión. Los comandos CLI de CUPS suelen necesitar privilegios de root.

Repasemos estos comandos:

- lpc: Permite varias formas de control sobre las impresoras. Con estado lpc, puede ver una lista de colas disponibles y el estado de cada una.
- lpstat: Muestra una lista de trabajos en cola de impresión en las impresoras del sistema. Puede usar varias opciones para poder modificar la salida de este comando.
- lpq: Muestra el estado de la cola actual o de la cola especificada con la opción –P.

- lppasswd: Cambia la contraseña de CUPS usada por el sistema. Establece AuthType como Digest en el archivo de configuración `cupsd.conf`.
- enable y disable: Inicia o para la cola especificada. El comando usado con mayor frecuencia es disable con la opción –c, que para una cola y cancela todos los trabajos de dicha cola.
- accept and reject: Hace que la cola de impresión empiece a aceptar o a rechazar nuevos trabajos.
- lprm: Elimina un trabajo de la cola. Puede especificar la cola (-P cola) y el identificador del trabajo (obtenido mediante lpstat).
- lpmove: Mueve un trabajo de impresión desde una cola a otra con identificador de trabajo y un nombre de cola (por ejemplo lpmove queue1-46 queue2).

Puede introducir estos comandos usted mismo. Aquí le mostramos un ejemplo de una impresora que hemos configurado usando la interface CUPS:

```
# lpc status
BrotherHL1440:
   printer is on device 'parallel' speed -1
   queuing is enabled
   printing is enabled
   no entries
   daemon present
```

Gestión de usuarios

En Linux, puede gestionar usuarios (añadir, cambiar, eliminar) de muchas formas. Al comienzo de la sección, vamos a asumir que cada servidor que tenga que administrar tiene su propia base de datos de usuarios, ubicada en el archivo `/etc/passwd`.

También vamos a asumir que conoce los aspectos básicos de añadir y eliminar cuentas de usuario con los comandos adduser y useradd para cualquier distribución que use, puesto que varían según la distribución.

Las distribuciones Linux han cambiado el comportamiento por defecto de los comandos adduser/useradd. Puede consultar las páginas de manual para ambos comandos. Tendrá que experimentar cómo se comporta su distribución. En Fedora, los dos comandos parecen comportarse igual: ambos añaden una cuenta y crean un directorio de usuario. Si teclea adduser tadelste o useradd tadelste, ambos comandos añadirán al usuario y crearán su directorio personal, pero no le pedirán una contraseña temporal o le harán algunas preguntas estándar que quizá cabría esperar.

En otras distribuciones, podría ver una salida como esta:

```
... # adduser tadelste
Adding user 'tadelste'...
Adding new group 'tadelste' (1001).
Adding new user 'tadelste' (1001) with group 'tadelste'.
Creating home directory '/home/tadelste'.
Copying files from '/etc/skel'
Enter new UNIX password: passwd1
Retype new UNIX password: passwd1
passwd: password updated successfully
Changing the user information for tadelste
Enter the new value, or press ENTER for the default
   Full Name []: New User
   Room Number []:
   Work Phone []: 999-555-1212
   Home Phone []:
   Other []:
Is the information correct? [y/N] y
```

En Fedora, no obstante, la salida se para en la línea "Copiando archivos..." Luego, el administrador puede introducir la primera contraseña del usuario. Pero, ¿qué pasa si el administrador no asigna inmediatamente una contraseña a un nuevo usuario? ¿Podría el nuevo usuario acceder al servidor vía ssh, por ejemplo? Veámoslo:

```
$ ssh tadelste@host2.centralsoft.org
tadelste@host2.centralsoft.org's password:
Permission denied, please try again.
tadelste@host2.centralsoft.org's password:
Permission denied, please try again.
tadelste@host2.centralsoft.org's password:
Permission denied (publickey,gssapi-with-mic,password).
$
```

Como puede ver, la respuesta es no. El usuario no sólo no tiene una contraseña en blanco; sino que no tiene contraseña. El archivo `ssh_config` tiene activado el requisito de contraseña, por lo que el usuario no podrá acceder vía SSH.

El usuario root debe proporcionar una contraseña para el usuario, para lo que puede hacer lo siguiente:

```
[root@host2 ~]# passwd tadelste
Changing password for user tadelste.
New UNIX password: passwd1
Retype new UNIX password: passwd1
passwd: all authentication tokens updated successfully.
[root@host2 ~]#
```


La salida indica que el comando passwd ha cambiado la contraseña del usuario, pero no es así; no ha preguntado por la (inexistente) contraseña original. Como usuario, una vez que se le haya asignado una contraseña, podrá cambiarla usted mismo:

```
$ passwd
Changing password for user tadelste.
Changing password for tadelste
(current) UNIX password: passwd1
New UNIX password: passwd1
Password unchanged
New UNIX password: passwd2
Retype new UNIX password: passwd2
passwd: all authentication tokens updated successfully.
$
```

Fedora primero verifica que tiene una contraseña (de no ser así, no podrá autentificarse en el servidor). También verifica que la nueva contraseña que ha introducido es distinta de la ya existente. Si introduce la misma contraseña, Fedora no la acepta y le pide una nueva.

Puesto que Fedora usa el protocolo Red Hat, necesitará saber que existen algunos aspectos de seguridad a la hora de añadir usuarios u opciones de contraseñas.

Cuando instaló Fedora, el script de instalación le pidió que proporcionara una contraseña para la cuenta de root y que creara de manera opcional una cuenta de usuario además de la de root. Además de eso, sólo debe tener un poco de experiencia añadiendo usuarios y quizás en la administración de grupos.

Los administradores de sistemas necesitan saber:

- Cómo crear y definir cuentas.
- Cómo eliminar y desactivar cuentas.
- La amenaza que suponen los exploits de seguridad a la hora de gestionar usuarios y cómo combatirla.

Debería saber que las cuentas de usuarios sirven para múltiples propósitos en los sistemas Linux, y que algunos "usuarios" no son personas. Veamos los tipos principales de cuentas:

- Cuentas para personas reales.

  A cada usuario se le proporciona una cuenta que está asociada con varias opciones de configuración, tales como una contraseña, un directorio personal y una shell que se ejecuta cuando el usuario se autentifica. Ofrecer cuentas separadas para cada usuario permite a las personas establecer los permisos de sus archivos, por lo que puede controlar quién accede a ellos.
- Cuentas para servicios del sistema, tales como servidores de correo o bases de datos.

Estas cuentas aseguran que los servicios se ejecutan con privilegios muy restringidos y tienen acceso sólo a los archivos necesarios, en caso de que errores de programación o intrusos malintencionados los creen e intenten afectar a otras partes del sistema. Normalmente, cuando se instala un servicio, el proceso de instalación o el administrador del sistema crea un usuario y un grupo con el mismo nombre (postfix, mysql, etc.) y le asigna todos los archivos y directorios controlados por el servicio. A los servicios no se les dan contraseñas, ni directorio personal ni Shell, debido a que los intrusos podrían usarlos.

Como se mencionó anteriormente, si está leyendo este libro, debería saber cómo añadir usuarios, establecer contraseñas, etc. Ahora, vamos a fijarnos en los aspectos que un administrador necesita conocer sobre los usuarios desde el punto de vista de la seguridad.

Eliminando a un usuario

En muchas ocasiones, los empleados duran poco tiempo en sus puestos. Por lo que a menos que administre una pequeña tienda con una base de empleados estable, necesitará aprender cómo eliminar un usuario después de que se haya ido.

Demasiados administradores de sistemas no comprenden los riesgos que conlleva gestionar usuarios. Los empleados disgustados a menudo suelen provocar problemas serios para una compañía si mantienen el acceso a la red.

Eliminar a un usuario no es un proceso de un único paso, involucra muchos archivos de usuarios, buzones de correo, alias de correo, trabajos de impresión, procesos personales (como el encargado de hacer copias de seguridad o el de sincronización remota de usuarios) y otros aspectos. Es una buena idea empezar por desactivar la cuenta de usuario en `/etc/passwd`; después de ello, puede buscar archivos del usuario y otras referencias.

Una vez que todas las trazas del usuario se hayan borrado, puede eliminar al usuario completamente (si elimina la entrada de `/etc/passwd` mientras existan referencias al usuario, lo pasará mal). Cuando elimina un usuario, es una buena idea seguir una serie preestablecida de pautas para así no olvidar ningún paso; puede hacerse una lista o puede programar un script que lo haga. La primera tarea es desactivar la contraseña de usuario, así se cortará el acceso de manera efectiva. Puede hacer esto con:

```
# passwd -l tadelste
```

Algunas veces es necesario desactivar temporalmente una cuenta sin eliminarla. Por ejemplo, un usuario tiene una baja por maternidad o se va a un viaje de 90 días a otro país. Cuando revise los logs del sistema puede descubrir que alguien ha conseguido acceso sin autorización adivinando su contraseña. El comando passwd -l es útil para estas situaciones.

Luego, tiene que decidir qué hacer con los archivos de usuario. Recuerde que hay usuarios que tienen archivos fuera de su directorio personal. El comando find puede encontrarlos:

```
# find / -user tadelste
[root@host2 ~]# find / -user tadelste
/home/tadelste
/home/tadelste/.zshrc
/home/tadelste/.bashrc
/home/tadelste/.bash_profile
/home/tadelste/.gtkrc
/home/tadelste/.bash_logout......
```

Luego ya puede decidir entre mantener estos archivos o eliminarlo. Si decide eliminarlos, haga una copia de seguridad por si se necesitaran después.

Como seguridad extra, puede cambiar la shell de autentificación del usuario por un valor sin sentido. Simplemente cambie el último campo del archivo passwd por `/bin/false`.

Si su organización usa shell seguras como SSH (a menudo se ofrece el Open SSHServer junto con Linux) y permite la autentificación remota mediante claves RSA o DSA, un usuario puede conseguir acceso al sistema incluso si la contraseña está desactivada, esto se debe a que SSH usa claves separadas.

Por ejemplo, incluso una vez que haya desactivado la contraseña de Tom Adelstein, él puede coger otro ordenador y ejecutar el siguiente comando:

```
$ ssh -f -N -L8000:intranet.yourcompany.com:80 my.domain.com
```

Esto redirige el tráfico al puerto 80 (el puerto donde suele escuchar el servidor Web) de su servidor interno. Obviamente, si su sistema ofrece SSH, debería eliminar las claves autorizadas de los directorios adecuados (por ejemplo `.~tadelste/.ssh` o `.~tadelste/.ssh2`) para evitar que el usuario vuelva a conseguir acceso a su cuenta:

```
$ cd .ssh
:~/.ssh$ ls
authorized_keys known_hosts
:~/.ssh$ rm authorized_keys
:~/.ssh$ ls
known_hosts
:~/.ssh$
```

Asimismo, busque los archivos .shosts y .rhosts en el directorio personal del usuario (por ejemplo, ~tadelste/.shosts y ~tadelste/.rhosts). También compruebe si el usuario tiene algún proceso ejecutándose en el sistema. Tales procesos

podrían servir como puerta trasera y permitirían al usuario acceder a la red. El siguiente comando le indica si un usuario tiene procesos ejecutándose:

```
# ps aux |grep -i ^tadelste
```

Algunas otras preguntas que un administrador de sistemas debería hacerse acerca de un usuario que ha abandonado la compañía serían:

- ¿Podría el usuario ejecutar scripts CGI desde su directorio personal o desde alguno de los servidores Web de la compañía?
- ¿Existe algún archivo de reenvío de correo tal como ~tadelste/.forward? Los usuarios pueden usar reenviadores de correo para sus cuentas, lo que puede provocar problemas a un sistema que se supone que les permite el acceso.

Sellando el directorio personal

A menudo se dará cuenta de que el administrador necesita mantener la información del directorio personal del usuario que se fue. Todo el correo y otros documentos de la cuenta personal del usuario pertenecen al usuario. En caso de que el empleado y la compañía se vean enfrentados en un juicio, tal vez pudiera ser necesario acceder a estos archivos. Muchos analistas consideran que mantener los directorios personales es una buena práctica. Puede guardar los contenidos del directorio de usuario renombrándolo. Simplemente ejecute el comando move:

```
# mv /home/tadelste /home/tadelste.locked
```

Esto evita que un usuario pueda autentificarse o hacer uso de archivos como el archivo .forward que se discutió en la sección previa. El contenido se conserva intacto en el caso de que se necesitara después.

Gestores gráficos de usuarios

A medida que Linux se fue introduciendo en el Mercado se fue haciendo mayor, compañías como Sun Microsystems, Novell, Computer Associates, HP e IBM empezaron a portar sus paquetes administrativos a Red Hat, SUSE y otras plataformas Linux. Además, las herramientas administrativas incluidas en las distribuciones Linux empezaron a madurar, lo que incrementó tanto la funcionalidad como la usabilidad.

Puesto que ahora ya tiene cierto conocimiento de los comandos y los procesos necesarios para limpiar una cuenta de usuario personal, puede descubrir utilidades

que son más fáciles de usar. Aunque normalmente descubrirá que son menos flexibles que usar la línea de comandos. Echemos un vistazo a una de estas herramientas, originalmente desarrollada para SUSE, se llama YaST2.

El configurador del Escritorio Java de Sun se muestra en la figura 8.14. La descripción de las funciones que puede realizar con la herramienta se muestra en el panel de la izquierda.

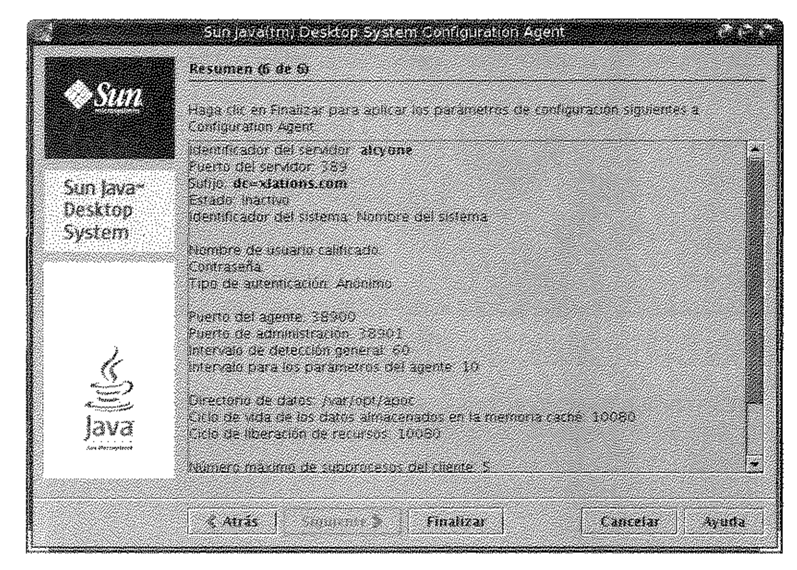


Figura 8.14. Gestor de usuarios JDS de Sun Microsystem.

Fíjese en que el cuadro de diálogo en la parte de arriba le pregunta si quiere eliminar el directorio /home/taldeste. Como se dijo previamente, su compañía puede desear mantener los directorios personales de los empleados que se han ido. En este caso, la herramienta gráfica le da sólo dos opciones: o borrar el directorio o no borrarlo. No le ofrece la opción de renombrar el directorio, que como se comentó antes, era la forma más segura de proceder.

En la figura 8.15, puede ver otro ejemplo tomado de un sistema Fedora.

Con la herramienta gráfica de gestión de usuarios de Fedora, puede realizar las mismas operaciones básicas que con la que se mostró en la figura 8.14. Pero tampoco ofrece todas las opciones que necesita para gestionar adecuadamente las cuentas de los usuarios que se fueron.

Aunque técnicamente no se trata de un gestor de usuarios, Fedora ofrece otra herramienta que se puede usar para configurar servicios relacionados con los usuarios. Vea la figura 8.16, la herramienta gráfica ofrecida por Fedora cuando introduce el comando.

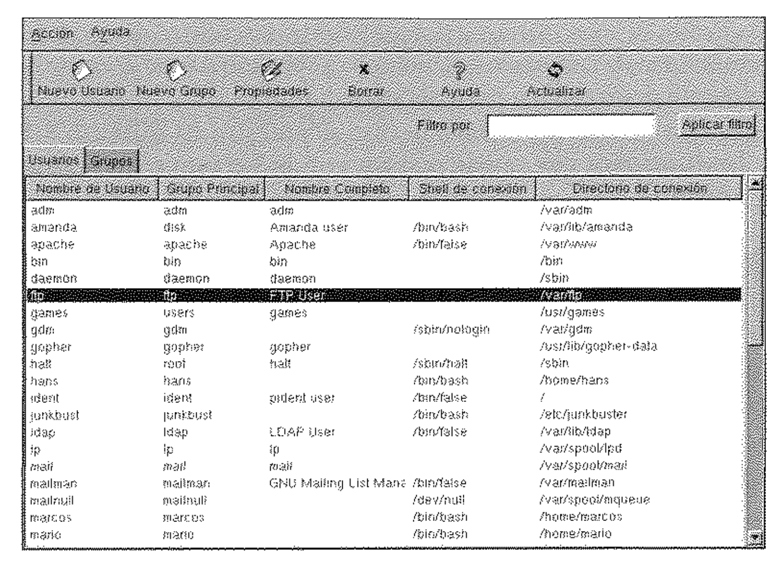


Figura 8.15. Herramienta gráfica de gestión de usuarios de Fedora.

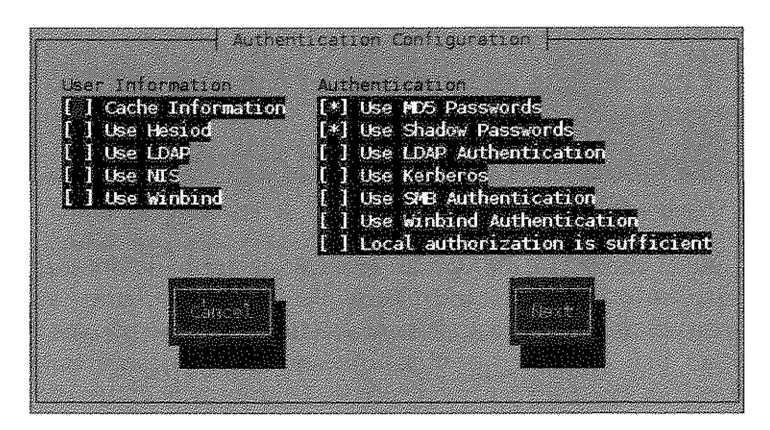


Figura 8.16. Configurador de autentificación Red Hat.

Este es otro ejemplo de las múltiples formas en que Linux permite gestionar las cuentas de usuario. No necesita que ejecute el sistema X Window.

Capítulo 9

Virtualización en la empresa moderna



En este capítulo, vamos a repasar un área que empiece a popularizarse dentro de la administración de sistemas Linux. Hoy en día, la virtualización juega un papel crucial en la consolidación de los centros de datos, la computación de alto rendimiento, el abastecimiento rápido, la continuidad comercial y la gestión de la carga de trabajo. Las empresas están viendo que la virtualización puede hacerles ahorrar en costes, y los analistas están notando que la tecnología está cambiando el paisaje de los negocios.

La virtualización es un concepto que ha ganado popularidad gracias a la exitosa compañía VMware (`http://www.vmware.com`) y el proyecto de código abierto Xen (`http://www.cl.cam.ac.uk/research/srg/netos/xen`). Se refiere a una máquina que ejecuta varios núcleos (que algunas veces son el mismo y otras veces pertenecen a diferentes sistemas operativos) y encima una capa muy fina de software que gestiona el acceso al hardware. Cada núcleo se llama invitado y actúa como si todo el procesador fuera suyo.

Los diferentes invitados son mucho más independientes con respecto a los otros que si fueran procesos independientes en un mismo sistema operativo. Este aislamiento ofrece seguridad y robustez debido a que el fallo de uno de los invitados no afecta a los otros. La capa de virtualización realiza muchas de las funciones de un sistema operativo, gestionando el acceso al procesador, los dispositivos y la memoria para cada invitado.

En la actualidad, los distintos desarrolladores Linux están trabajando en un nuevo sistema llamado Kernel Virtual Machine (KVM), que formará parte del núcleo.

Por qué la virtualización es tan popular

Para comprender quién está usando la virtualización y el entorno en la que es valiosa, debería comprender las necesidades actuales en el mundo de los negocios. Esta sección ofrece una introducción antes de comenzar a explicar cómo funciona la virtualización en Linux.

Todo el campo relacionado con las tecnologías de la información ha crecido de manera exponencial desde que aparecieron los primeros sistemas de archivos distribuidos. Las organizaciones han visto cómo sus infraestructuras se expandían año tras año. Mucha culpa de este crecimiento lo tiene la constante mejora de los componentes informáticos y del software. Pero no todo son ventajas.

Las tecnologías informáticas han evolucionado desde la gestión de transacciones hasta la gestión de los procesos de negocio. Algunas corporaciones están especializadas en la gestión de recursos humanos, otras en las finanzas y en la contabilidad, otras lo están en las manufacturas y en las cadenas de suministro. Esta especialización ha creado una relación de dependencia entre los centros de datos y el personal dedicado a las tecnologías de la información.

Las redes tradicionales ahora son capaces de capturar y gestionar diferentes tipos de transacciones que antes no podían, y ha surgido la necesidad de incrementar la potencia computacional y por tanto la capacidad de almacenamiento. Este crecimiento también ha tenido lugar en la forma de almacenar datos, para lo que se ha creado el término deslocalización de servidores (véase la figura 9.1).

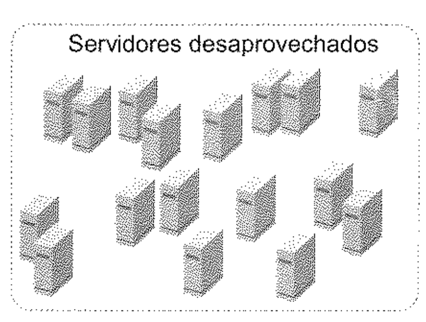


Figura 9.1. Granja de servidores deslocalizados, con un sistema operativo por torre.

Vayamos un paso más allá: hay aplicaciones especializadas para campos como el de la contabilidad y el de las finanzas que necesitan ejecutarse de manera separada, se necesitan pues servidores con una gran disponibilidad y con hardware redundantes para asegurar la continuidad comercial. Esta combinación de factores ha transformado el paisaje de las TIC en una combinación de servidores físicos demasiado grande y aislada, y donde muchos recursos están infrautilizados.

Buena parte de la culpa la tienen las normas del mercado, que provoca que los gastos se incrementen: tiene que aumentar su capacidad de almacenamiento para recuperar documentos, que en muchos casos deberá conservar hasta más de 25 años.

De no ser así, sus sucesores no tendrán la tecnología para generar toda esa información, que por otra parte puede ser muy útil para abogados o auditores, por ejemplo.

Echemos otro vistazo a los resultados del crecimiento de la informática:

- Los servidores y las aplicaciones con una única función (normalmente llamados "silos") tienen su capacidad infrautilizada.
- Los costes adiciones se incrementan debido a la complejidad del software y a la necesidad de manejar grandes cantidades de datos.
- La necesidad de personal especializado en áreas funcionales donde existe una carencia de documentación y grandes niveles de incompetencia.
- La necesidad de guiar y dar soporte a los usuarios y a los administradores además de mantener actualizado el software.

Ahora ya puede hacerse una idea de por qué la virtualización empresarial ha ganado popularidad y ha pasado a ser una de las áreas donde la tecnología está cambiando el paisaje de los negocios.

Con las imágenes virtuales, puede fácilmente comprimir los datos junto con todos los programas, opciones de configuración, librerías del sistema operativo y otros metadatos que conforman el sistema entero. Restaurar una imagen hace que el sistema vuelva a quedar exactamente igual a como estaba cuando se hizo la imagen, lo que facilita la reproducción de los documentos. La virtualización tiene los siguientes beneficios:

- Sustituye enormes conjuntos de sistemas que apenas se usan por unos cuantos sistemas mejor utilizados.
- Simplifica la administración, porque separa los núcleos con una aplicación ejecutándose en cada uno, aumentando así la seguridad y facilidad de gestión. También mantiene un entorno en el que los documentos que se generan mantienen una serie de requisitos.
- Reduce el hardware y la complejidad, por lo que las plantillas de personal pueden ser más pequeñas.
- La virtualización puede ayudar a invertir la tendencia de infrautilización de los servidores.

Computación de alto rendimiento

Linux se ha convertido en el sistema operativo preferido para alojar máquinas virtuales debido a su capacidad para ejecutar y gestionar clusters y grids de PC.

Los grandes vendedores de hardware tardaron en comprenderlo, pero una vez que se convencieron, vieron en ello un jugoso negocio. Durante varios años, Linux ha visto como grandes benefactores han contribuido al avance de su tecnología en un loable esfuerzo de desarrollo. Entre estos contribuidores están IBM, Intel, AMD, Novell, Red Hat, Unisys, Fujitsu y muchos otros.

Por ejemplo, IBM necesitaba un sistema operativo para su iniciativa OpenPower. De repente, Linux sacó su Big Blue Virtualization Engine en la forma de un hipervisor de código abierto. El motor de IBM permite crear y gestionar particiones y asignarles dinámicamente recursos de entrada/salida.

Los desarrolladores del núcleo de Linux han anunciado una nueva tecnología multi-hilo (SMT) e hiper-hilo. Linux ahora puede hacer que dos hilos se ejecuten de manera simultánea en el mismo procesador, esta tecnología es esencial para equipos que van a alojar sistemas operativos invitados. Además, VMware se ejecuta bien en Linux y ofrece una capa de virtualización para otras instancias de Linux u otros sistemas operativos. El User Mode Linux (UML) es otro ejemplo de virtualización para Linux.

La versión 2.6 del núcleo de Linux funciona bien con la tecnología SMT de IBM. Antes de esta versión, Linux no tenía una buena planificación de hilos ni un buen arbitraje. El núcleo 2.6 solucionó este problema e hizo crecer el número de procesadores en los que podía ejecutarse dicho núcleo.

Esto es importante por dos razones. Primero, como equipo que soporta máquinas virtuales, Linux tiene que realizar de manera eficiente la gestión de su hardware. Segundo, como los invitados son independientes del hardware físico, tiene que mantener su capacidad de manejar varios procesos a la vez. Hoy en día, Linux hace ambas cosas de manera extraordinaria. Gestiona el hardware y el particionamiento virtual y se ejecuta bien en las particiones invitadas gracias a HP y a IBM.

Si alguna vez se ha preguntado por qué compañías como XenSource y VirtualIron han aparecido de la nada, ahora ya lo sabe: debido a sus contribuciones al hipervisor de código abierto. Al igual que los vendedores de hardware se percataron de que podían aumentar las ventas de PC y de componentes, los vendedores de software también pensaron que podían aumentar sus ventas. Incluso Microsoft se dio cuenta de que necesitaba entrar en el negocio de Linux, por lo que contribuyó a XenSource y a Virtual Iron.

Continuidad comercial y gestión de la carga de trabajo

Incluso a pequeña escala, su organización se beneficiará de separar el correo electrónico, DNS, servidores Web y directorios, pasarelas y bases de datos. Colocar cada uno de estos servicios en un único servidor asegura que si un servidor se cae, no cae también la infraestructura entera. Pero separar los servicios del

hardware físico requiere mucho tiempo, espacio, dinero y gastos indirectos. También necesitará hacer copias de seguridad y restaurar datos, prever catástrofes y hacerse con el hardware que mejor se ajusta a su trabajo.

Con la virtualización Linux, puede hacer dividir un único servidor físico en un grupo de varios servidores virtuales. Cada servidor virtual es como si fuera un servidor físico para los administradores de sistemas. Puede crear una instancia de servidor separada para cada servicio que quiera ofrecer: correo electrónico, DNS, servidor Web, etc. Si uno falla, los otros no lo notarán.

Al dividir el equipo físico también se posibilita la creación de diferentes configuraciones para cada servidor virtual, a pesar de que comparten el mismo hardware. En un entorno, por ejemplo, hemos creado máquinas virtuales más pequeñas (VM) para nuestros servidores DNS y otras más grandes para el correo y el servidor Web. Esto nos permite repartir la carga de trabajo y mantener el mismo hardware. La figura 9.2 le da una idea de lo que puede hacer con un único servidor físico.

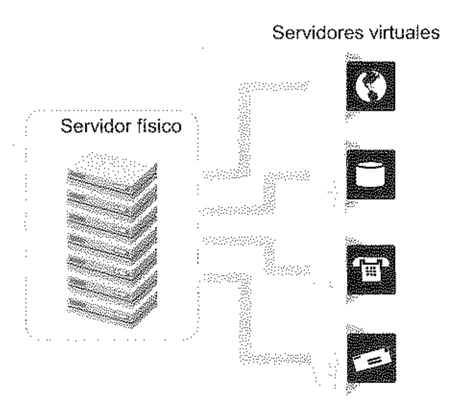


Figura 9.2. Dividiendo un único servidor físico en múltiples máquinas virtuales.

Abastecimiento rápido

Primero hemos hecho la virtualización en nuestra red creando una instalación mínima de Debian en VM. Una vez que hemos cubierto nuestras necesidades, la hemos comprimido y la hemos grabado en un CD. Luego, hemos configurado nuestras máquinas virtuales adicionales usando VMware con distintas configuraciones y hemos copiado la imagen comprimida en cada directorio especificándolo a VM.

También hemos configurado las máquinas virtuales Xen usando Fedora con instalaciones mínimas. Luego, hemos añadido los componentes que necesitábamos para cada servicio que queríamos ofrecer. Por ejemplo, nuestro servidor DNS

primario se ejecuta en una máquina virtual Xen, mientras nuestros servidores de correo electrónico y de páginas Web se ejecutan en instancias separadas de VMware.

Nota: Cada VM reside en un directorio. Por ejemplo, nuestro directorio principal `/var/lib/vmware/Virtual Machines` contiene varios subdirectorios como `debian-31r0a-i386-netinst-kernel2.6`. Simplemente hemos comprimido ese subdirectorio y lo hemos usado para la creación de otros subdirectorios con nombres ligeramente diferentes.

Después de que tenemos un servidor (por ejemplo de correo) ejecutándose, hemos hecho una copia comprimida y la hemos volcado a un CD. De manera regular y sistemática hacemos copias de seguridad de cada servidor volcándolas a CD o DVD. También hemos intentado mover las imágenes a distribuciones Linux distintas y se han ejecutado como esperábamos.

Cómo ayuda la virtualización

¿Qué decíamos que aportaba la virtualización? Primero, hemos eliminado la necesidad de varios servidores físicos. Hemos convertido nuestro sistema operativo preferido en una imagen, por lo que solo hemos necesitado lanzar el proceso de instalación una vez. Luego, hemos creado máquinas virtuales y hemos copiado sistemáticamente nuestras imágenes virtuales para permitir una recuperación rápida en caso de que el sistema fallara.

La virtualización funciona bien para pequeñas compañías, a las que permite levantar su infraestructura con software libre. Imagine los gastos en licencias que se han evitado. Ahora, imagine qué tipos de estrategias pueden adoptar las grandes compañías usando Linux.

Llegados a este punto, quizás tenga ganas de aprender a realizar este trabajo. Por lo que vamos a empezar por el proceso de instalación y configuración de Xen y VMware y hacer una demostración de cómo virtualizar una red de servidores.

Instalando Xen en Fedora 5

En esta sección, vamos a mostrar cómo instalar Xen en una única máquina para gestionar dos sistemas operativos. Puede que algún día Xen convierta esta forma en el estándar de las distribuciones Linux y haga así más fácil las instalaciones. Pero por ahora, se necesita una labor manual:

Estamos usando Fedora Core 5 (FC5) como el sistema operativo que aloja Xen, puesto que soporta Xen 3.0. Preguntemos a Yum (un gestor de paquetes similar al apt-get de Debian y al up2date de Red Hat) sobre Xen:

```
# yum info xen
Loading "installonlyn" plugin
Setting up repositories
core                                                        [1/3]
updates                                                     [2/3]
extras                                                      [3/3]
Reading repository metadata in from local files
Available Packages
Name : xen
Arch : i386
Version: 3.0.2
Release: 3.FC5
Size : 1.4 M
Repo : updates
Summary: Xen is a virtual machine monitor
Description:
This package contains the Xen hypervisor and Xen tools, needed to
run virtual machines on x86 systems, together with the kernel-xen*
packages. Information on how to use Xen can be found at the Xen
project pages.

Virtualisation can be used to run multiple versions or multiple
Linux distributions on one system, or to test untrusted applications
in a sandboxed environment. Note that the Xen technology is still
in development, and this RPM has received extremely little testing.
Don't be surprised if this RPM eats your data, drinks your coffee
or makes fun of you in front of your friends.
```

Esto parece alentador. Vamos a probarlo, pero primero revisemos los requisitos:

- El sistema debe tener por lo menos 256 MB de RAM.
- Su gestor de arranque debe ser grub.
- SELINUX debe ser disable o permissive, pero no enforcing.

Ejecute el programa system-config-securitylevel o edite `/etc/selinux/config` que debería ser así:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - SELinux is fully disabled.
SELINUX=Disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#     targeted - Only targeted network daemons are protected.
#     strict - Full SELinux protection.
SELINUXTYPE=targeted
```

Si cambia el valor SELINUX por enforcing, necesitará reiniciar Fedora antes de proceder.

Este comando instalará el hipervisor Xen, un núcleo Fedora modificado por Xen y llamado dominio 0, además de varias utilidades:

```
# yum install kernel-xen0
```

Nota: La necesidad de un núcleo de Linux modificado por Xen desparecerá en un futuro cuando Intel y AMD introduzcan soporte para virtualización en sus chips. También se espera que Windows Vista soporte la virtualización a nivel de procesador.

Esto añade xen0 como primera elección en el archivo /boot/grub/grub.conf, pero no como núcleo por defecto:

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to
this file
# NOTICE:  You have a /boot partition. This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/VolGroup00/LogVol00
#          initrd /initrd-version.img
#boot=/dev/hda
default=1
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora Core (2.6.17-1.2157_FC5xen0)
   root (hd0,0)
   kernel /xen.gz-2.6.17-1.2157_FC5
   module /vmlinuz-2.6.17-1.2157_FC5xen0 ro root=/dev/VolGroup00/LogVol00
   module /initrd-2.6.17-1.2157_FC5xen0.img
title Fedora Core (2.6.17-1.2157_FC5)
   root (hd0,0)
   kernel /vmlinuz-2.6.17-1.2157_FC5 ro root=/dev/VolGroup00/LogVol00
   initrd /initrd-2.6.17-1.2157_FC5.img
title Fedora Core (2.6.15-1.2054_FC5)
   root (hd0,0)
   kernel /vmlinuz-2.6.15-1.2054_FC5 ro root=/dev/VolGroup00/LogVol00
   initrd /initrd-2.6.15-1.2054_FC5.img
default=0
```

Para hacer que Xen sea el núcleo por defecto, cambie esta línea:

```
default=1
```

por:

```
default=0
```


Ahora ya puede reiniciar. Xen debería iniciarse automáticamente, veámoslo:

```
# /usr/sbin/xm list
Name                               ID Mem(MiB) VCPUs State  Time(s)
Domain-0                           0     880       1 r----     20.5
```

La salida debería mostrar que Domain-0 se está ejecutando. Domain 0 controla todos los sistemas operativos invitados que se ejecutan en el procesador, de manera análoga a como el núcleo controla los procesos de un sistema operativo.

Instalando sistemas operativos invitados en Xen

Xen es ahora quien controla el procesador, pero necesitará añadir al menos un sistema operativo invitado. Empezaremos instalando Fedora Core 5 como invitado, porque facilita la labor, y luego ofreceremos algunos consejos para otras variantes de Linux.

Fedora Core 5

Fedora Core 5 tiene un script de instalación como invitado de Xen que facilita el proceso, aunque sólo instala invitados FC5. El script espera acceder al árbol de instalación de FC5 vía FTP, Web o NFS; por algunas razones, no puede especificar un directorio o un archivo. Usaremos nuestro DVD de instalación de FC5 que serviremos a través de Apache.

```
# mkdir /var/www/html/dvd
# mount -t iso9660 /dev/dvd /var/www/html/dvd
# apachectl start
```

Ahora vamos a ejecutar el script de instalación y a responder a sus preguntas:

```
# xenguest-install.py
What is the name of your virtual machine? guest1
How much RAM should be allocated (in megabytes)? 256
What would you like to use as the disk (path)? /xenguest
What is the install location? http://127.0.0.1/dvd
```

Llegados a este punto, comienza la instalación de FC5. Elija entre el modo texto o el modo gráfico (si se está ejecutando X) vía vnc. Si elige modo texto, se conectará a una consola. Proceda como lo haría normalmente con una instalación de Fedora o de Red Hat. En la pantalla de dirección IP, proporcione al invitado una dirección distinta que la del equipo, si usa DHCP (si escogió dhcp="dhcp" en el archivo de configuración de Xen, como se explica en la siguiente sección). La última pantalla le preguntará si quiere reiniciar. Desmonte el DVD y extráigalo. Ahora se reiniciará sólo el nuevo sistema invitado, no Xen ni el equipo.

Xen no inicia el sistema operativo invitado automáticamente. Necesita introducir este comando en el equipo:

```
# xm create guest1
```

Llegados a este punto, tendremos dos sistemas operativos (host1 y guest1) funcionando independientemente y viviendo en armonía, cada uno con sus propios sistemas de archivos, conexiones de red y memoria. Para probar que ambos servidores se están ejecutando, pruebe estos comandos:

```
# xm list
Name                         ID Mem(MiB) VCPUs State  Time(s)
Domain-0                     0     128       1 r----    686.0
guest1                       3     256       1 -b----   14.5
# xentop
xentop -'21:04:38 Xen 3.0-unstable
2 domains: 1 running, 1 blocked, 0 paused, 0 crashed, 0 dying, 0 shutdown
Mem: 982332k total, 414900k used, 567432k free CPUs: 1 @ 2532MHz
   NAME STATE CPU(sec) CPU(%) MEM(k) MEM(%) MAXMEM(k) MAXMEM(%)VCPUS NETS
NETTX(k) NETRX(k) SSID
Domain-0 ----r 686  0.3   131144   13.4  no limit    n/a      1  8
1488528     80298    0
guest1 --b--   14   0.1   261996   26.7  262144   26.7        1  1
129       131      0
```

Para que los dominios de Xen se inicien automáticamente, use estos comandos:

```
# /sbin/chkconfig --level 345 xendomains on
# /sbin/service xendomains start
```

Otros invitados

Si quiere instalar un sistema operativo invitado distinto de FC5, necesitará editar el archivo de configuración de Xen, que es un archivo de texto (actualmente, un script de Python) en el directorio /etc/xen.

xmexample1 y xmexample2 son archivos de ejemplo comentados. Para ver la sintaxis del archivo completo, teclee:

```
# man xmdomain.cfg
```

Cuando ejecutamos xenguest-install.py en la sección previa, se generó una configuración de invitado de Xen /etc/xen/guest1, con algunas líneas extra:

```
# Automatically generated Xen config file
name = "guest1"
memory = "256"
disk = [ 'file:/xenguest,xvda,w' ]
vif = [ 'mac=00:16:3e:63:c7:76' ]
```



```
uuid = "bc2c1684-c057-99ea-962b-de44a038bbda"
bootloader="/usr/bin/pygrub"

on_reboot = 'restart'
on_crash = 'restart'
```

Esto contiene algunas, pero no todas, de las directivas que el invitado necesita. Un archivo de configuración mínimo para el invitado debería quedar así:

1. Un nombre único de dominio invitado:

   ```
   name="vm01"
   ```

2. Una ruta para la imagen del núcleo del dominio invitado:

   ```
   kernel="/boot/vmlinuz-2.6.12.6-xenU"
   ```

3. Un dispositivo raíz para el dominio invitado:

   ```
   root="/dev/hda1"
   ```

4. Reserva de memoria inicial para el invitado, en megabytes:

   ```
   memory=128
   ```

Nota: La suma de la memoria para todos los invitados de Xen no debe exceder la memoria física, además hay que tener muy en cuenta que Xen usa 64 MB.

---

5. El espacio en disco para el dominio invitado. Esto está definido en uno o más dispositivos de bloque, cada uno debe ir entre comillas simples:

   ```
   disk = [ 'stanza1', 'stanza2' ]
   ```

   Donde los dispositivos se representan mediante una cadena de tres parámetros ('host_dev, guest_dev, mode'). host_dev es el área de almacenamiento del domino tal y como es vista por el equipo. Puede ser:

   - file:pathname
     Un archivo de imagen (un archivo local que Xen tratará como sistema de archivos); este archivo se crea cuando ejecuta el programa run create o xen-create-image.
   - phy:device
     Un dispositivo físico.

   guest_dev es el dispositivo físico tal y como lo ve el dominio invitado, y los modos son o bien *r* para lectura o *w* para lectura y escritura. Por ejemplo, una directiva de disco de ejemplo para dos invitados es:

   ```
   disk=['file:/vserver/images/vm01.img, hda1, w', 'file:/vserver/
   images/vm01-swap.
   img, hda2, w']
   ```

6. La interfaz de información de red es una directiva vif. Esta directiva debe contener la especificación de un dispositivo de red. La red por defecto se especifica con:

```
vif=[ ' ' ]
```

Una directiva dhcp controla si se usa DHCP o se usa la interfaz de información. Lo siguiente especifica el uso de DHCP:

```
dhcp="dhcp"
```

Si la directiva dhcp no aparece o está definida como "off", debe especificar la información de red de manera estática, hágalo cuando configure el sistema:

```
ip="192.168.0.101"
netmask="255.255.255.0"
gateway="192.168.0.1"
hostname="vm01.example.com"
```

La ayuda de xm proporciona el siguiente ejemplo de un invitado mínimo, con un archivo de imagen del equipo apareciendo como dispositivo raíz del invitado:

```
kernel = "/boot/vmlinuz-2.6-xenU"
memory = 128
name = "MyLinux"
root = "/dev/hda1 ro"
disk = [ "file:/var/xen/mylinux.img,hda1,w" ]
```

Una vez que tiene el archivo de configuración del invitado, cree un invitado de Xen con este comando:

```
# xm create -c guest_name
```

donde `guest_name` puede ser una ruta completa o un nombre de archivo relativo (en cuyo caso Xen lo coloca en `/etc/xen/guest_name`). Xen creará el dominio invitado e intentará arrancarlo desde el dispositivo o el archivo dado. La opción –c adjunta una consola al dominio al iniciarse, por lo que puede responder a las preguntas que el proceso de instalación le haga.

Instalando VMware

VMware ofrece su servidor gratuitamente, e incluso el código se distribuye bajo la licencia de código abierto. Puede encontrarlo en `http://www.vmware.com/products/server`. Verá que es robusto y amigable. Puede leer más acerca del código de VMware y su comunidad de iniciativas de código abierto en ese mismo sitio Web.

Como mencionamos anteriormente, arranques como el de XenSource y Virtual Iron se benefician del soporte que el núcleo de Linux ofrece a la tecnología de

hipervisión de IBM. En una dura competición con Xen, VMware también ha enviado sus propias propuestas de código abierto a los desarrolladores del núcleo, destacando el hecho de que VMware se ejecuta mejor en Linux si es el propio WMware el que proporciona una pequeña ayuda al núcleo.

Mientras ejecutamos Xen usando Fedora Core 5, hemos decido instalar VMware en un servidor Ubuntu y como sistema operativo invitado Debian. También hemos gestionado instancias remotas de WMware del escritorio Ubuntu usando la consola WMware. Luego, hemos instalado FC5 en la máquina virtual VMware.

Hemos descargado Vmware-server-1.0.1-29996.tar.gz y lo hemos descomprimido en un directorio de instalación llamado vmware-server-distrib. Dentro del directorio hemos encontrado vmwareinstall.pl y lo hemos ejecutado con `./vmware-install.pl`. Al poco tiempo, el programa de instalación ha empezado y ha mostrado los siguientes mensajes:

```
Creating a new installer database using the tar3 format.

Installing the content of the package

In which directory do you want to install the binary files?
[/usr/bin]
```

La instalación del servidor VMware comienza con varias preguntas como esta, que se basan en información relativa al sistema operativo y a los archivos de configuración capturada por el script de instalación.

Durante el proceso de instalación, el script le pedirá que acepte la licencia del producto VMware. Debería leerla antes de aceptar. Después de que acepte la licencia, VMware verifica el compilador y que los archivos de cabecera de su sistema sean compatibles unos con otros para obtener el programa VMware binario usando el compilador. Verá mensajes del tipo:

```
The path "/usr/lib/vmware" does not exist currently. This program is going
to create it, including needed parent directories. Is this what you want?
[yes]
```

Adicionalmente, verá compilaciones de código como las del siguiente ejemplo:

```
make[1]: Entering directory '/usr/src/linux-headers-2.6.15-26-k7'
CC [M]   /tmp/vmware-config0/vmnet-only/driver.o
CC [M]   /tmp/vmware-config0/vmnet-only/hub.o
CC [M]   /tmp/vmware-config0/vmnet-only/userif.o
CC [M]   /tmp/vmware-config0/vmnet-only/netif.o
CC [M]   /tmp/vmware-config0/vmnet-only/bridge.o
CC [M]   /tmp/vmware-config0/vmnet-only/procfs.o
CC [M]   /tmp/vmware-config0/vmnet-only/smac_compat.o
SHIPPED  /tmp/vmware-config0/vmnet-only/smac_linux.x386.o
LD [M]   /tmp/vmware-config0/vmnet-only/vmnet.o
```

```
Building modules, stage 2.
MODPOST
```

Hacia el final de la instalación, el script le informará que la compilación del código ha terminado y le ofrecerá un comando que puede usar para desinstalar el servidor:

```
The installation of VMware Server 1.0.1 build-29996 for Linux completed
successfully. You can decide to remove this software from your system at any
time by invoking the following command: "/usr/bin/vmware-uninstall.pl".
```

El script de instalación también le pedirá que ejecute el comando de configuración de la siguiente forma:

```
Before running VMware Server for the first time, you need to configure it by
invoking the following command: "/usr/bin/vmware-config.pl". Do you want
this program to invoke the command for you now? [yes]
```

Cuando el proceso de instalación finalice totalmente, podrá ver los siguientes mensajes:

```
Starting VMware services:
Virtual machine monitor                                  done
Virtual Ethernet                                         done
Bridged networking on /dev/vmnet0                        done
Host-only networking on /dev/vmnet1 (background)         done
Host-only networking on /dev/vmnet8 (background)         done
NAT service on /dev/vmnet8                               done
Starting VMware virtual machines                         done
The configuration of VMware Server 1.0.1 build-29996 for Linux for this
running kernel completed successfully.
```

Puede descargar una imagen que ya exista de un sistema operativo, lo que VMware llama aplicación desde la dirección `http://www.vmware.com/vmtn/appliances/directory`. Nosotros hemos escogido `debian-31r0a-i386-netinst-kernel2.6.zip` que está ubicada en el directorio `/var/lib/vmare/Virtual Machines` y la hemos descomprimido.

Una vez que ya tenemos nuestra imagen básica, hemos iniciado una consola de gestión de VMware desde un equipo remoto con Ubuntu de escritorio y detrás del cortafuegos de la ubicación remota. Hemos ejecuta el comando:

```
$ gksu vmware-server-console
```

Luego, hemos configurado la consola para conectar nuestro sistema operativo remotamente. Con la consola servidor de VMware ejecutándose, nos hemos conectado a la máquina virtual remota y nos hemos autentificado como root, como se muestra en la figura 9.3.

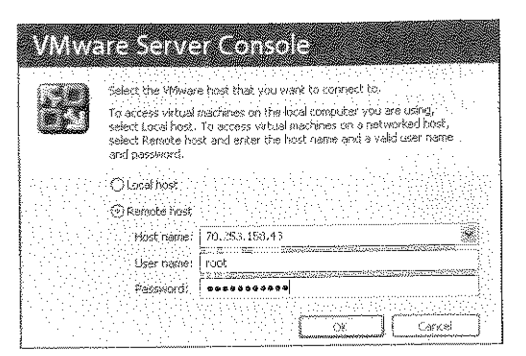


Figura 9.3. Conectando con un host virtual remoto.

Después de que nos hemos conectado al host remoto, VMware nos ha preguntado si queríamos crear una máquina virtual. Debido a que ya hemos creado una, hemos hecho clic en el menú Archivo y hemos abierto el directorio que contiene la instancia de Debian. Esta acción ha añadido Debian al inventario de VM. Nuestra consola luego se ha vuelto parecida a la de la figura 9.4, lo que nos da una idea de las funciones operativas disponibles.

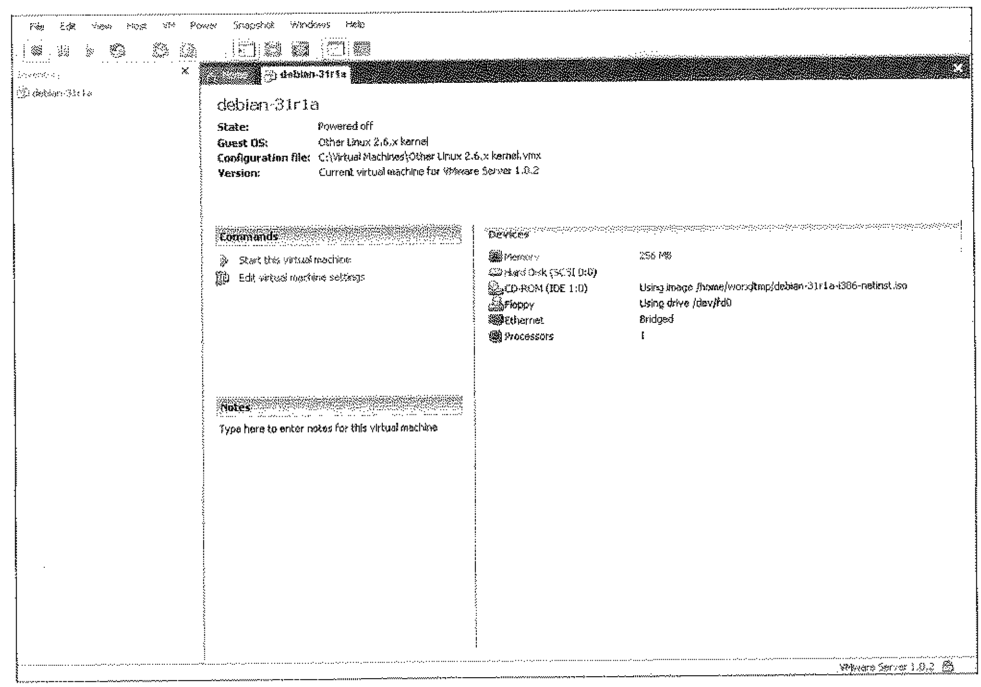


Figura 9.4. Conectado a un equipo remoto y listo para empezar.

Hemos sido capaces de iniciar Debian. Cuando el sistema ha arrancado, Debian ha empezado a ejecutar las últimas fases del script de instalación. Hemos permitido que se ejecutase, y en un breve lapso de tiempo hemos obtenido la pantalla de la figura 9.5.

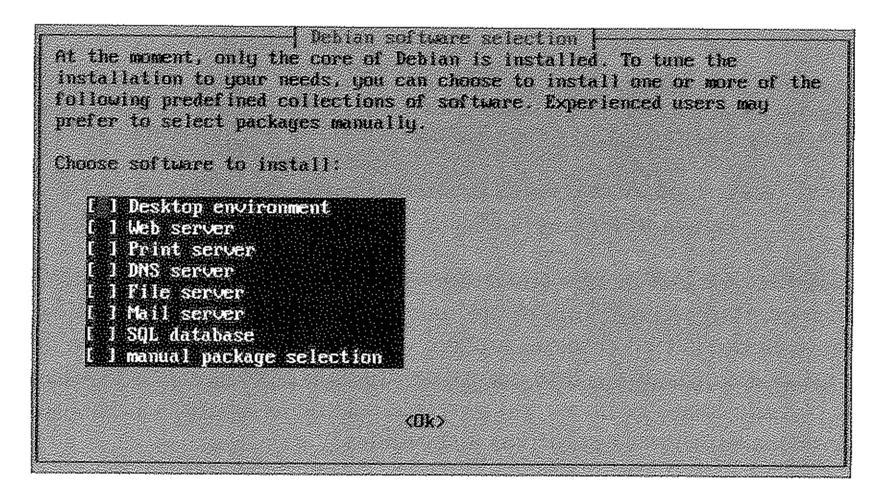


Figura 9.5. El script de instalación de Debian se ejecuta bajo una máquina virtual remota.

Hemos optado por configurar Debian manualmente en lugar de escoger una configuración predefinida. Esto nos permite crear un servidor Debian por defecto para implantar instancias adicionales del servidor VMware. La figura 9.6 muestra cómo se ejecuta Debian en el sistema.

La captura de pantalla nos muestra cómo se está ejecutando el comando ipconfig. Hemos probado esta instancia para asegurarnos de que las tarjetas Ethernet virtuales apuntaban correctamente a las direcciones IP que hemos configurado.

Una vez que tenemos nuestra imagen básica de Debian, la hemos comprimido y la hemos volcado a un CD. Luego, hemos implantando esa imagen en otros equipos, después hemos determinado el papel de cada sistema invitado y los requisitos en cuanto a recursos.

La figura 9.7 ofrece un resumen de la imagen Debian. En la parte derecha de la pantalla, puede ver la configuración del equipo. Podemos alterar el servidor virtual dinámicamente para añadir memoria, espacio en disco, tarjetas Ethernet, procesadores y varios dispositivos a medida que las necesidades se incrementen y tengamos que configurar máquinas adicionales.

Instalando sistemas operativos invitados en VMware

Para nuestra tarea final, es decir instalar otro sistema operativo, hemos descargado Fedora Core 5 desde el sitio de la comunidad VMware, la hemos copiado

al directorio de máquinas virtuales y la hemos descomprimido como hicimos con Debian. Luego, la hemos añadido a través del menú Archivo. La figura 9.8 muestra una pregunta sobre un identificador único, puede mantener el existente.

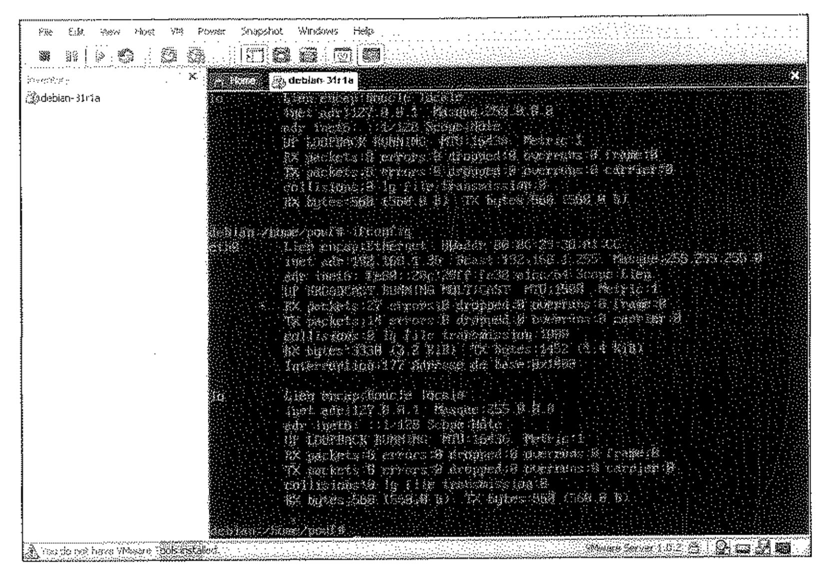


Figura 9.6. La instancia instalada de Debian en su equipo remoto.

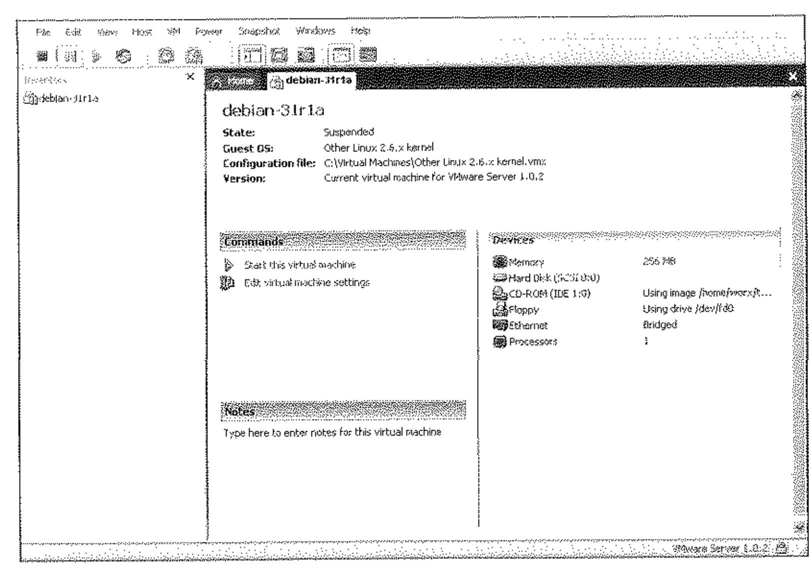


Figura 9.7. Sumario de consola de nuestra imagen Debian invitada.

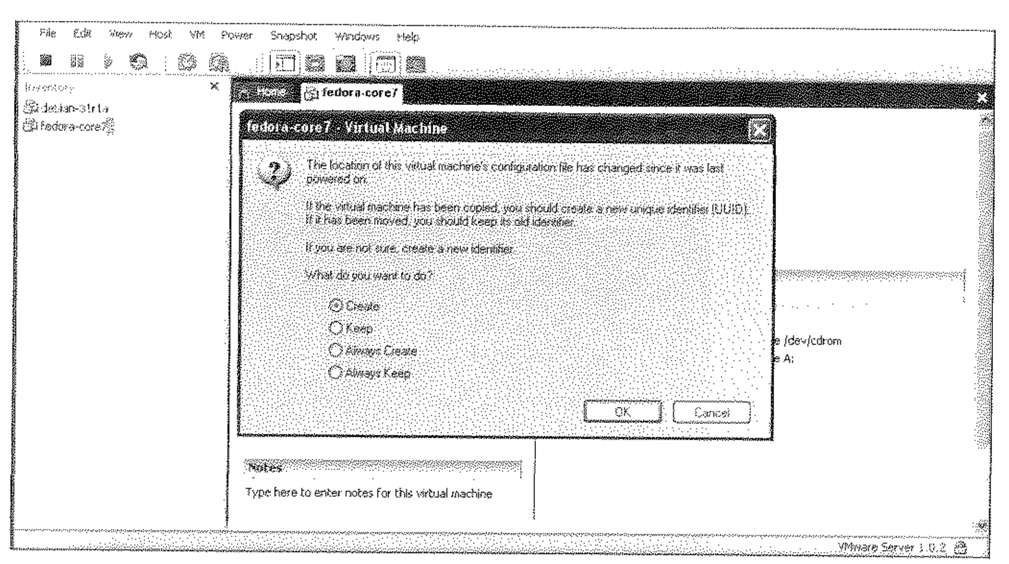


Figura 9.8. VMware pregunta sobre el identificador único de la imagen de una máquina virtual.

La consola de gestión de VMware se percata de que hemos añadido una imagen. Para distinguir entre posibles múltiples imágenes, hemos preguntado por el identificador único (UUID) en el diálogo que se muestra en la figura 9.8. Debido a que hemos copiado Fedora 5 y tenemos todos los archivos comprimidos en la imagen, no importa la opción del cuadro de diálogo que escojamos. Cuando abre una nueva máquina virtual, VMware le da la posibilidad de verificar la configuración virtual del hardware. La figura 9.9 le da una idea del inventario de hardware virtual disponible para Fedora Core 5.

Además de descargar imágenes y cargarlas en la consola de gestión, puede instalar un sistema operativo Linux desde un CD de distribución estándar.

Virtualización, ¿una moda pasajera?

Muchos analistas dicen que la virtualización Linux pasará de moda. Como administrador de sistemas, debe sopesar las ventajas y las desventajas de especializarse en esta tecnología. La virtualización no es el equivalente a la introducción del PC por parte de IBM, o a la introducción de los sistemas de archivos distribuidos por parte de Microsoft. El impacto de la tecnología de hipervisión no puede compararse a los programas ERP tales como SAP, PeopleSoft u Oracle Financials.

En cualquier caso, tecnologías como Xen y VMware tienen beneficios incuestionables. La virtualización mejora la utilización de los servidores y reduce

las adquisiciones desmesuradas de hardware, aprovechando mejor los recursos del sistema. Al ejecutar su actual software en un entorno virtual, puede sacar mejor partido a su inversión, puesto que son servidores de bajo coste sin por ello dejar de ser estándares.

Afortunadamente, este capítulo le ha ofrecido el conocimiento y las destrezas que necesita para implementar sus propios entornos virtuales. Ahora tiene la oportunidad de experimentar y divertirse con la tecnología de virtualización. Haciendo eso podrá posicionarse como especialista en un campo en el que muy pocos comprenden.

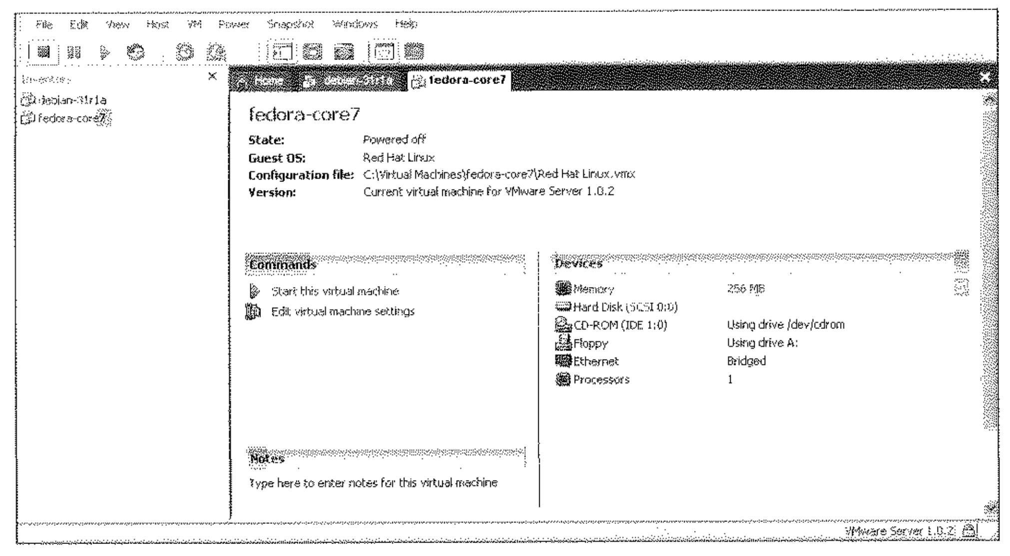


Figura 9.9. Configuración virtual del hardware para Fedora Core 5.

Capítulo 10

Scripting



Como administrador de sistemas Linux, usará dos herramientas con más frecuencia que el resto: un editor de texto para crear y editar archivos de texto, y una shell para ejecutar comandos. Llegados a un cierto punto, empezará a estar cansado de introducir comandos repetitivos y buscará formas de cuidar sus dedos y reducir errores. Aquí será cuando tendrá que combinar el editor de texto con la Shell para crear los programas Linux más simples: los shell scripts.

Linux usa por sí mismo scripts para casi todo, especialmente para tareas personalizables como la gestión de servicios y de procesos. Si comprende cómo están escritos estos scripts, puede interpretar los pasos que hay que tomar para adaptarlos a sus propias necesidades.

La shell (una intèrfaz del sistema operativo) es una de las muchas innovaciones heredadas del bisabuelo de Linux, Unix. En 1978, El investigador de los Laboratorios Bell Stephen Bourne desarrolló la Bourne Shell para la versión 7 de Unix. Se llamó sh (Unix valoraba la concisión), y definió las características estándar que hasta el día de hoy presentan todas las shell. Las shells evolucionaron desde su comienzo, amparadas en el desarrollo de la Korn Shell (ksh por supuesto), por la C Shell (csh) y por la Bash Shell (bash) que ahora mismo es la estándar de los sistemas GNU/Linux. Bash es un acrónimo de Bourne-Again Shell, y soporta scripts que fueron escritos originalmente para la Bourne Shell.

Este capítulo comienza con los aspectos básicos de la bash: los shell prompts, los comandos y parámetros, las variables, las expresiones, la redirección de E/S. Si ya está familiarizado con esto, no se perderá mucho si se salta unas cuantas páginas (excepto quizás una cura de insomnio). Cada herramienta tiene sus límites, y en muchas ocasiones puede resultar que bash no sea la mejor solución a sus problemas. Al final del capítulo examinaremos una pequeña aplicación escri

ta en varios lenguajes de script: bash, Perl, PHP y Python (las tres P están asociadas con la P del acrónimo que mencionamos en capítulos anteriores). Puede comparar su estilo, sintaxis, expresividad, facilidad de uso y aplicabilidad a diferentes dominios. No todos los problemas serán tan pequeños, pero pueden servir de ejemplo.

Comenzando con bash

Muchos sistemas operativos ofrecían interfaces de línea de comandos en los primeros días, y normalmente ofrecían la posibilidad de almacenar los comandos en archivos de texto y ejecutarlos como programas de procesamiento por lotes (un concepto que penetró muy bien en aquel tiempo). Pronto se hizo normal la forma de enviar parámetros a los scripts y de hacer que cambiaran su comportamiento en función de diferentes condiciones. La Shell de Unix hizo grandes avances en cuanto a flexibilidad se refiere, convirtiéndose en un verdadero lenguaje de programación.

Nuestros ejemplos interactivos mostrarán un ejemplo de shell prompt, un comando con parámetros opcionales y la salida del comando. Aquí se lo mostramos:

```
admin@server1:~$ date
Thu Aug 24 09:16:56 CDT 2006
```

Mostraremos el contenido de un shell script como este:

```
#!/bin/bash
contents of script...
```

La primera línea es una línea especial: si comienza con los dos caracteres #!, el resto de la primera línea es el nombre del comando que ejecutará como proceso el resto del script. (Si el carácter # no va seguido por !, se interpretará como un comentario que continua hasta el final de la línea). Este truco le permite usar cualquier programa para interpretar sus scripts. Si el programa es una Shell tradicional como sh o bash, el archivo se Shell script. Al final del capítulo le mostraremos scripts para Perl, PHP y Python.

Nota: *Microsoft Windows usa la extensión del archivo para definir el tipo* de archivo y qué intérprete debería ejecutarlo. Si cambia la extensión, deja de funcionar. En Linux, los nombres de los archivos no tienen nada que ver con la ejecución (aunque suelen seguirse algunas convenciones útiles por otras razones).

Use su editor de texto favorito (o incluso uno que no le guste) para crear este archivo de tres líneas y guárdelo en un archivo llamado hello:


```
#!/bin/bash
echo hello world
echo bonjour monde
```

Este archivo no es un script que funcione todavía. Le enseñaremos la forma de ejecutarlo en la siguiente sección, pero antes necesitamos explicar algunas reglas sintácticas básicas.

La Shell /bin/bash interpretará el script línea por línea. Espera que cada comando esté en una única línea, pero si termina una línea con una barra inclinada (\), bash tratará la siguiente línea como una continuación:

```
#!/bin/bash
echo \
hello\
world
```

Esto supone una buena forma de hacer las líneas complejas más legibles.

La Shell ignora las líneas con espacios en blanco (espacios, tabulaciones, líneas vacías). También ignora todo lo que vaya desde el carácter # hasta el final de la línea. Cuando bash lee la segunda línea del script (echo hello world), trata la primera palabra (echo) como si fuera el comando a ejecutar y el resto de palabras (hello world) como los parámetros. El comando echo sólo copia los argumentos a la salida. La tercera línea también lee otro comando echo, pero con distintos parámetros.

Para ver qué hay en el archivo hello, puede imprimir todo su contenido en la pantalla:

```
admin@server1:~$ cat hello
#!/bin/bash
echo hello world
echo bonjour monde
```

Rutas y permisos

El archivo hello puede ejecutarse lanzando el comando bash con el argument hello:

```
admin@server1:~$ bash hello
hello world
bonjour monde
admin@server1:~$
```

Ahora intentemos ejecutar hello sin poner bash delante:

```
admin@server1:~$ hello
bash: hello: command not found
```

¿Por qué bash no puede encontrarlo? Cuando especifica un comando, Linux busca en una lista de directorios la ruta de un archivo con ese nombre y ejecuta el primero que encuentre. En este caso, hello no estaba en estas listas de directorios. Si le indica al sistema en qué directorio está hello, lo ejecutará. La ruta puede ser absoluta (/home/admin/hello) o relativa (./hello significa que el archivo hello está en este directorio). Describiremos cómo especificar los directorios en su ruta en la próxima sección, pero antes tenemos que tratar los permisos.

Una shell script no se ejecutará sin ciertos permisos de archivo. Comprobemos los permisos de hello:

```
admin@server1:~$ ls -l hello
-rw-r--r-- 1 admin admin 48 2006-07-25 13:25 hello
```

Un – indica que la bandera no está activa. El primer – es la bandera de directorio, es una d para los directorios y un – para los archivos. Luego vienen los permisos para el propietario del archivo, luego los del grupo y por último, los de cualquiera. El propietario (admin) puede leer (r) y escribir (w) el archivo, mientras que el grupo (en este caso también se llama admin) y el resto de la gente sólo puede leerlo (r-). Nadie puede ejecutar el archivo, puesto que el tercer carácter de cada grupo de tres marca un – en lugar de una x.

Ahora, intentemos ejecutar hello con una ruta relativa:

```
admin@server1:~$ ./hello
bash: ./hello: Permission denied
```

Esta vez Linux lo encontró pero no lo ejecutó. Falló porque el archivo no tenía permisos de ejecución. Necesita decidir quién podrá ejecutarlo: sólo usted (el propietario), cualquiera del grupo, y/o los usuarios de otros grupos. Esto es una decisión de seguridad práctica que los administradores deben hacer con frecuencia. Si los permisos están muy repartidos, otros pueden ejecutar su script sin conocimiento; si son demasiado restrictivos, el script no se ejecutará.

El comando para cambiar los permisos se llama chmod (viene del inglés change mode) y puede usar el estilo de números octales de los antiguos Unix o letras. Intentémoslo de las dos formas, proporcionándose permisos de lectura/escritura/ejecución a usted mismo, lectura/escritura/ejecución al grupo y nada a los otros (¿qué comparten ellos con usted?). Para el estilo octal, lectura=4, escritura=2 y ejecución=1. El número de usuario será 4+2+1 (7), el grupo 4+1 (5) y otros 0:

```
admin@server1:~$ chmod 750 hello
admin@server1:~$ ls -l hello
-rwxr-x-- 1 admin admin 50 2006-08-03 15:44 hello
```

El otro tipo de parámetros de permisos es usar letras, lo que probablemente es más intuitivo:


```
admin@server1:~$ chmod u=rwx,g=rx hello
admin@server1:~$ ls -l hello
-rwxr-x-- 1 admin admin 50 2006-08-03 15:44 hello
```

Para añadirse permisos a sí mismo, a su grupo o a los otros de forma rápida, introduzca:

```
admin@server1:~$ chmod +xr hello
admin@server1:~$ ls -l hello
-rwxr-xr-x 1 admin admin 50 2006-08-03 15:44 hello
```

Ahora, podemos ejecutar el script desde la línea de comandos:

```
admin@server1:~$ ./hello
hello world
bonjour monde
```

La ruta por defecto

La lista de directorios en la cual bash debería buscar los comandos están especificados en una variable de entorno llamada PATH. Para ver qué hay en su PATH, introduzca:

```
admin@server1:~$ echo $PATH
/bin:/usr/bin
```

Linux reserva los nombres especiales . para el directorio actual y .. para el directorio padre actual. Si quiere que Linux siempre encuentre comandos como hello en su directorio actual, añada el directorio actual a PATH.

```
admin@server1:~$ PATH=$PATH:.
```

Para hacer cambios tales como éste, necesitará hacer un cambio permanente en PATH. Esto puede hacerse por un usuario individual en el archivo .bashrc ubicado en el directorio personal, o por el administrador del sistema en un archivo de configuración que abarca todo el sistema (normalmente ubicado en el directorio /etc); sólo añada una sentencia al archivo como el comando que hemos mostrado.

De forma alternativa, podría mover el script hello a uno de los directorios que ya están en el PATH. Sin embargo, estos directorios normalmente están protegidos, por lo que sólo el usuario root puede poner archivos allí, para mantener la seguridad.

Para un script más complejo que hello (casi todos los scripts), ambos métodos tienen implicaciones de seguridad. Si . es su PATH, corre el riesgo de que si alguien pone un script diferente llamado hello en otro directorio y usted comete el error de entrar en ese directorio y teclear hello, ejecutará el hello del otro usua-

rio y no es lo que quería ejecutar. La corrección del script también supone una preocupación. Estamos seguros de que sabe qué hace su script, pero no lo estaríamos tanto después de añadir cientos de líneas.

Una práctica común es poner sus propios script en un directorio como `/usr/local/bin` o `~/bin` en lugar de un directorio del sistema como `/bin`, `/sbin` o `/usr/bin`. Para añadir este directorio a su PATH de manera permanente, añada un línea como la siguiente al final de su archivo .bashrc:

```
export PATH=$PATH:/usr/local/bin
```

Redirección de E/S

La redirección de E/S y las tuberías son innovaciones Unix que Microsoft y muchos otros han copiado descaradamente. La Shell le da acceso a estas funcionalidades de manera muy intuitiva.

Cuando está tecleando un comando en la consola o en una ventana de texto, sus dedos son la entrada estándar de comandos y sus ojos leen la salida estándar o la salida de errores estándar. No obstante, puede producir entradas o capturar salidas en sustitución de sus dedos o sus ojos con un archivo. Ejecute el comando ls con la pantalla como salida estándar y luego rediríjala (con >) a un archivo:

```
admin@server1:~$ ls
hello
admin@server1:~$ ls > files.txt
admin@server1:~$
```

En el segundo ejemplo, la redirección sucede silenciosamente. Si ocurre algún error, no obstante, debería verlo en la pantalla en lugar del archivo (es decir, por qué se ha producido el error):

```
admin@server1:~$ ls ciao > files.txt
ls: ciao: No such file or directory
admin@server1:~$
```

Tenga en cuenta que si el archivo `files.txt` existe antes de ejecutar estos comando. Se sobrescribirá. Si quiere añadir nuevo contenido al archivo en lugar de sobrescribirlo, use los caracteres (>>):

```
admin@server1:~$ ls -l >> files.txt
```

Si `files.txt` no existe, se creará antes de que se produzca la anexión. También puede redirigir los errores estándares. Aquí se muestra un comando que redirige la salida estándar y la salida de errores estándar al mismo tiempo:

```
admin@server1:~$ ls -l > files.txt 2> errors.txt
```


El nada elegante 2> es la redirección de errores. La redirección de errores puede ser útil en procesos largos como compilaciones, por lo que puede revisar cualquier mensaje de error más tarde en lugar de buscarlos en la pantalla.

Si quiere redirigir la salida estándar y la salida estándar de errores al mismo archivo, haga esto:

```
admin@server1:~$ ls -l > files.txt 2>&1
```

El &1 significa "el mismo lugar que la salida estándar", que es el caso de `files.txt`. Un atajo para el comando anterior es:

```
admin@server1:~$ ls -l >& files.txt
```

Use >> en lugar de > donde quiera añadir en lugar de sobrescribir.

Sólo se suele redirigir la salida estándar. Aquí mostramos un ejemplo que busca los nombres de los archivos en los que está contenida la cadena foo:

```
admin@server1:~$ ls -l > files.txt
admin@server1:~$ grep foo < files.txt
admin@server1:~$ rm files.txt
```

El primer paso crea el archivo temporal `files.txt`. El segundo paso lo lee, y en el tercero somos limpios y lo borramos. La vida de los archivos temporales es corta pero productiva.

Podemos combinar estos tres tipos de pasos en uno y evitar el archivo temporal con la mejor invención de Unix, la tubería. Una tubería conecta la salida de un comando con la entrada de otro comando. El símbolo de la tubería es |, y también un > o < y consigue mucha velocidad. La salida estándar del primer comando se convierte en la entrada estándar del segundo comando, simplificando nuestros primeros pasos:

```
admin@server1:~$ ls -l | grep foo
```

También puede encadenar varias tuberías:

```
admin@server1:~$ ls -l | grep foo | wc -l
```

Este comando contará el número de veces que la cadena foo aparece en alguno de los archivos del directorio.

Variables

bash es un lenguaje de programación y los lenguajes de programación tienen funcionalidades comunes. Una de las más básicas es la variable: un símbolo que

contiene un valor. Las variables bash son cadenas a menos que especifique otro tipo con la sentencia "declare". No necesita declarar o definir variables bash antes de usarlas, como sucede en otros muchos lenguajes.

Un nombre de variable es una cadena que comienza con una letra y contiene letras, número o guiones bajos (_). El valor de una variable se obtiene poniendo el símbolo $ antes del nombre de la variable. Aquí mostramos un shell script que asigna un valor de cadena a la variable hw y luego la imprime:

```
#!/bin/bash
hw="hello world"
echo $hw
```

La variable hw se crea con la asignación de la línea 2. En la línea 3, los contenidos de la variable hw sustituyen a la referencia $hw. Debido a que bash y otras shell tratan los espacios en blanco (espacios y tabulaciones) como un separador de parámetros en lugar de cómo parámetros normales, para conservarlos debe encerrarlos con comillas dobles (") o simples ('). La diferencia es que las variables shell (y otra sintaxis especial) se expanden con comillas dobles y se tratan literalmente como comillas simples. Vea la diferencia en la salida de los dos comandos echo del siguiente script:

```
admin@server1:~$ cat hello2
#!/bin/bash
hw="hello world"
echo "$hw"
echo '$hw'
admin@server1:~$ ./hello2
hello world
$hw
admin@server1:~$
```

Puede asignar la salida estándar de un comando a una variable con la sintaxis $(comando) o 'comando' (usando las tildes):

```
admin@server1:~$ cat today
#!/bin/bash
dt=$(date)
dttoo=`date`
echo "Today is $dt"
echo "And so is $dttoo"
admin@server1:~$ ./today
Today is Tue Jul 25 14:56:01 CDT 2006
And so is Tue Jul 25 14:56:01 CDT 2006
admin@server1:~$
```

Las variables especial representan parámetros de línea de comandos. El carácter $ seguido por n parámetros de la línea de comandos, empezando por 1. La variable $0 es el nombre del script. La variable $* contiene todos los argumentos como un valor de cadena. Estas variables pueden pasarse junto con los comandos que el script ejecuta:


```
admin@server1:~$ cat files
#!/bin/bash
ls -Alv $*
admin@server1:~$ ./files hello hello2 today
-rwxr-xr-x 1 admin admin 48 2006-07-25 13:25 hello
-rwxr-xr-x 1 admin admin 51 2006-07-25 14:45 hello2
-rwxr-xr-x 1 admin admin 45 2006-07-25 14:49 today
admin@server1:~$
```

La variable especial $$ contiene el ID de proceso actual. Puede usarse para crear un nombre de archivo temporal único. Si se están ejecutando varias copias del mismo script al mismo tiempo, cada una tendrá un ID de proceso diferente y también un nombre de archivo temporal distinto.

Otra variable útil es $?, que contiene el valor que devolvió el comando ejecutado más recientemente. Usaremos esto más tarde en este capítulo para comprobar el éxito o el fallo en la ejecución de un programa desde un script.

Elementos útiles para bash Scripts

Hemos introducido los elementos básicos para bash que usará a diario a la hora de ejecutar comandos interactivos. Ahora, veamos algunas cosas que le ayudarán a escribir scripts efectivos.

Expresiones

Las expresiones bash contienen variables y operadores como == (igual) y > (mayor que). Estos actualmente se usan en comprobaciones, que pueden especificarse de varias formas:

```
test $file == "test"
[ $file == "test" ]
[[ $file == "test" ]]
```

Si usa el comando test, recuerde que algunos símbolos tienen múltiples significados (por ejemplo, en una sección anterior, hemos usado > para la redirección de la salida) por lo que necesitan ir entre comillas. No tiene que preocuparse sobre las comillas si usa el corchete o el doble corchete. Los dobles corchetes hacen lo mismo que uno sólo y un poco más, por lo que es seguro usar dobles corchetes con sus expresiones. bash tiene algunos operadores útiles:

```
-a file  # true si el archivo existe
-d file  # true si el archivo existe y es un directorio
-f file  # true si el archivo existe y es un archivo
-r file  # true si el archivo existe y es legible
-w file  # true si el archivo existe y se puede escribir
-x file  # true si el archivo existe y es ejecutable
```

Aritmética

bash está bastante inclinado al tratamiento de texto como comandos, parámetros y nombres de archivos. Puede evaluar las expresiones aritméticas normales (usando +, -, *, / y otros operadores) pero encerrándolos entre un par de paréntesis: ((expresión)). Debido a que muchos caracteres aritméticos –incluyendo *, (, y) se interpretan de manera especial por parte de la shell, es mejor entrecomillar los parámetros de las shell si se pretende que sean tratados como expresiones matemáticas en el script:

```
admin@server1:~$ cat arith
#!/bin/bash
answer=$(( $* ))
echo $answer
admin@server1:~$ ./arith "(8+1)*(7-1)-60"
-6
admin@server1:~$ ./arith "2**60"
1152921504606846976
admin@server1:~$
```

La última versión de bash soporta los enteros de 64 bits (desde –9223372036854775808 hasta 9223372036854775807). Las versiones más antiguas soportan sólo enteros de 32 bits (con un rango desde –2147483648 hasta 2147483647). Los números en coma flotante no se soportan. Los scripts que necesitan coma flotante u operadores más avanzados pueden usar un programa externo como bc.

En las expresiones aritméticas, puede usar variables con el carácter $ que debería usarse para sustituir sus valores por otras opciones:

```
admin@server1:~$ cat arithexp
#!/bin/bash
a=$1
b=$(( a+2 ))
echo "$a + 2 = $b"
c=$(( a*2 ))
echo "$a * 2 = $c"
admin@server1:~$ ./arithexp 6
6 + 2 = 8
6 * 2 = 12
admin@server1:~$
```


If...

Puede ejecutar diferentes porciones de código dependiendo del resultado de las comprobaciones. Bash usa la sintaxis if...fi (sentencia condicional) con elif como opcional y la sección else:

```
if expression1 ; then
   (commands)
elif expression2 ; then
   (commands)
...
elif expressionN ; then
   (commands)
else (commands)
fi
```

La construcción ; then al final de una línea también puede expresarse como un then plano en la línea siguiente:

```
if expression
then
   (commands)
fi
```

Si está en el mismo directorio que el script hello que hicimos antes, pruebe esto:

```
admin@server1:~$ if [[ -x hello ]]
> then
> echo "hello is executable"
> fi
hello is executable
admin@server1:~$
```

Aquí mostramos un script que busca en el archivo /etc/passwd un nombre de cuenta:

```
#!/bin/bash
USERID="$1"
DETECTED=$( egrep -o "^$USERID:" < /etc/passwd )
if [[ -n "${DETECTED}" ]] ; then
   echo "$USERID is one of us :-)"
else
   echo "$USERID is a stranger :-("
fi
```

Llame a este script friendforfoe, hágalo ejecutable y ejecútelo primero con una cuenta del sistema (root) y luego con otra cuenta (sasquatch):

```
admin@server1:~$ ./friendorfoe root
root is one of us :-)
```

```
admin@server1:~$ ./friendorfoe sasquatch
sasquatch is a stranger :-(
```

El primer parámetro se asigna a la variable shell USERID. El comando egrep se ejecuta con $() para asignar la salida a la variable shell detectada. egrep –o imprime sólo la cadena que coincide, en lugar de la línea entera. "^USERID:" coincide con los contenidos de la variable USERID sólo si los contenidos de la variable aparecen al inicio de una línea y vienen seguidos después por un punto. La expresión `if` está encerrada en corchetes dobles para contenerla, evaluarla y devolver el resultado. La expresión –n "${DETECTED}" devuelve true si la variable shell DETECTED no es una cadena vacía. Finalmente, la variable DETECTES está entrecomillada ("${DETECTED}") para tratarla como una siguiente cadena. Allí donde la sentencia `if` coge una expresión, puede poner un comando o una secuencia de comandos. Si el último comando de la secuencia tiene éxito, la sentencia `if` considera que la expresión ha devuelto un valor verdadero. Si el último comando de la secuencia falla. Se considera que la expresión devolvió un false, y se ejecutará la expresión else. Veremos ejemplos en las siguientes secciones.

Depurando un script sencillo

Vamos a seccionar un script que se supone que borra su parámetro (un archivo o un directorio) pero tiene algunos problemas:

```
admin@server1:~$ cat delete
#!/bin/bash
if rm $1
   then
   echo file $1 deleted
else
   if rmdir $1
      then
      echo directory $1 deleted
   fi
fi
```

El script está intentando eliminar el archivo que se le ha pasado como parámetro usando rm e imprimir un mensaje diciendo que se ha logrado. Si rm falla, el script asume que el parámetro se refiere a un directorio e intenta ejecutar el comando rmdir.

Aquí mostramos algunos resultados:

```
admin@server1:~$ ./delete hello2
file hello2 deleted
admin@server1:~$ ./delete hello2
rm: cannot remove 'hello2': No such file or directory
```



```
rmdir: 'hello2': No such file or directory
admin@server1:~$ mkdir hello3
admin@server1:~$ ./delete hello3
rm: cannot remove 'hello3': Is a directory
directory hello3 deleted
admin@server1:~
```

Usando estos mensajes de error, vamos a intentar arreglar el script. Primero, vamos a usar redirección de E/S para guardar los resultados en un log y en archivos de error que podemos revisar cuando tengamos un hueco. Luego, capturaremos el valor del comando rm para generar un mensaje de logro o de fallo. También capturaremos la fecha y la hora actual para incluirlos en el log de salida:

```
admin@server1:~$ cat removefiles
#!/bin/bash
# removefiles deletes either files or directories
echo "$0 ran at" $(date) >> delete.log
if rm $1 2>> delete-err.log
   then
   echo "deleted file $1" >> delete.log
elif rmdir $1 2>> delete-err.log
   then
   echo "deleted directory $1" >> delete.log
else
   echo "failed to delete $1" >> delete.log
fi
```

El script todavía tiene algunos fallos: no comprueba si el archivo existe y no distingue entre archivos y directorios. Podemos usar algunos operadores de los ya mencionados para arreglar estos problemas:

```
admin@server1:~$ cat removefiles
#!/bin/bash
# removefiles deletes either files or directories
echo "$0 ran at" $(date) >> delete.log
if [ ! -e $1 ]
   then
   echo "$1 does not exist" >> delete.log
elif [ -f $1 ]
   then
   echo -n "file $1 " >> delete.log
   if rm $1 2>> delete-err.log
      then
      echo "deleted" >> delete.log
   else
      echo "not deleted" >> delete.log
   fi
elif [ -d $1 ]
   then
   echo "directory $1 " >> delete.log
```

```
    if rmdir $1 2>> delete-err.log
       then
       echo "deleted" >> delete.log
    else
       echo "not deleted" >> delete.log
    fi
fi
```

Esto parece muy bonito, pero todavía hay más dificultades: ¿qué pasa si el archivo o el directorio contiene espacios? (Está garantizado que los verá en sistemas como Windows o Mac.) Cree un archivo llamado my file y luego intente eliminarlo con nuestro script:

```
admin@server1:~$ ./removefiles my file
```

La última línea de `delete.log` contendrá:

```
my does not exist
```

Puesto que no hemos entrecomillado el nombre del archivo, la shell dividirá my y file en dos variables $1 y $2. Por lo que debemos intentar mantener my file en $1:

```
admin@server1:~$ ./removefiles "my file"
./removefiles: [: my: binary operator expected
./removefiles: [: my: binary operator expected
```

Ahora hemos incluido la cadena my file en la variable shell $1, pero necesitamos quitarle las comillas dentro del script para que las comprobaciones y las eliminaciones funcionen bien:

```
admin@server1:~$ cat removefiles
#!/bin/bash
# removefiles deletes either files or directories
echo "$0 ran at" $(date) >> delete.log
if [ ! -e "$1" ]
   then
   echo "$1 does not exist" >> delete.log
elif [ -f "$1" ]
   then
   echo -n "file $1 " >> delete.log
   if rm "$1" 2>> delete-err.log
      then
      echo "deleted" >> delete.log
   else
      echo "not deleted" >> delete.log
   fi
elif [ -d "$1" ]
   then
   echo -n "directory $1 " >> delete.log
   if rmdir "$1" 2>> delete-err.log
      then
      echo "deleted" >> delete.log
   else
      echo "not deleted" >> delete.log
   fi
fi
```


Ahora, por último, cuando ejecute el comando:

```
admin@server1:~$ ./removefiles "my file"
```

la última línea de `delete.log` será:

```
file my file deleted
```

Bucles

Si quiere hacer algo más de una vez, necesita un bucle. bash tiene tres tipos de bucles: `for`, `while` y `until`.

El bucle `for` es amigable y muy potente, tiene esta apariencia general:

```
for arg in list
do
   commands
done
```

Ejecuta una acción (que puede ocupar tantas líneas y comandos como quiera) especificada entre do y done para cada elemento de una lista. Cuando los comandos se ejecutan, pueden acceder al elemento actual de la lista a través de la variable $arg. La sintaxis puede ser un poco confusa al principio: en la sentencia `for` debe especificar arg sin el símbolo del dólar, pero en los comandos debe especificar $arg con dicho símbolo.

Algunos ejemplos sencillos son:

```
admin@server1:~$ for stooge in moe larry curly
> do
> echo $stooge
> done
moe
larry
curly

admin@server1:~$ for file in *
> do
> ls -l $file
> done
```

```
-rw-r--r-- 1 admin admin 48 2006-08-26 14:12 hello

admin@server1:~$ for file in $(find / -name \*.gif)
> do
> cp $file /tmp
> done
```

El bucle loop se ejecuta mientras que la condición de prueba sea cierta:

```
while expression
do
   stuff
done
```

Aquí le mostramos un script de ejemplo que usa expresiones aritméticas que ya se han mencionado para crear un bucle `while` al estilo C (las sangrías no son necesarias, pero nos gustan):

```
#!/bin/bash
MAX=100
((cur=1)) # Treat cur like an integer
while ((cur < MAX))
   do
   echo -n "$cur "
   ((cur+=1)) # Increment as an integer
done
```

El bucle `until` es el opuesto al `while`. Itera hasta que la condición de prueba sea cierta:

```
until expression
do
   stuff
done
```

Un ejemplo es:

```
#!/bin/bash
gameover="q"
until [[ $cmd == $gameover ]]
   do
   echo -n "Your commmand ($gameover to quit)? "
   read cmd
   if [[ $cmd != $gameover ]]; then $cmd; fi
   done
```

Para escapar de un bucle, use break. Vamos a reescribir nuestro ejemplo de `until` como un bucle `while` con un break:

```
#!/bin/bash
gameover="q"
```



```
while [[ true ]]
   do
   echo -n "Your commmand ($gameover to quit)? "
   read cmd
   if [[ $cmd == $gameover ]]; then break; fi
   $cmd
   done
```

Para saltarse el resto del bucle y volver al inicio, use continue:

```
#!/bin/bash
gameover="q"
while [[ true ]]
   do
   echo -n "Your commmand ($gameover to quit)? "
   read cmd
   if [[ $cmd != $gameover ]]; then $cmd; continue; fi
   break
   done
```

Tareas cron

Los shell scripts a menudo se usan para juntar programas. Un ejemplo muy común en Linux es la definición de tareas cron. cron es el planificador de tareas estándar de Linux. Si quiere que algo suceda el tercer martes de cada mes a la 1:23, puede hacer que una tarea cron lo haga por usted y así ahorrarse la malhumorada respuesta que obtendría por parte de un ser humano. El demonio cron comprueba cada minuto si es hora de hacer algo, o si alguna especificación de tarea cron ha cambiado. Puede especificar las tareas cron editando el archivo crontab. Puede ver el contenido de su crontab, si es que hay algo, de esta forma:

```
admin@server1:~$ crontab -l
no crontab for admin
```

Para editar su crontab, introduzca:

```
admin@server1:~$ crontab -e
```

Cada línea de un archivo crontab contiene la especificación del día/hora y un comando en este formato:

```
minuto hora día_del_mes mes día_de_la_semana comando
```

Esto requiere algo más que una pequeña explicación:

- minuto es un valor entre 0 y 59.
- hora usa el reloj de 24 horas y es un valor entre 0 y 23.

- día_del_mes es un valor entre 1 y 31.
- mes es un número entre 1 y 12 o un nombre como Febrero.
- día_de_la_semana es un número entre 0 y 7 (0 o 7 es domingo, 6 es sábado) o un nombre como martes.
- día_del_mes y día_de_la_semana funcionan como una OR, lo que puede provocar sorpresas.
  Por ejemplo, si cada campo contiene un 1, cron ejecutará el comando en Enero, pero también los lunes. Normalmente, la línea crontab contiene un solo valor específico en estos campos.
- En un campo, un valor significa una coincidencia exacta; por ejemplo, un 1 en el campo mes significa sólo Enero.
- Un asterisco (*) significa cualquier valor.
- Dos valores separados por un guión indican un rango. Es decir, 11-12 en el campo mes significa desde Noviembre hasta Diciembre.
- Para especificar más de un valor, separe los valores con comas. Una lista de meses 2,3,5-6 significa Febrero, Marzo y desde Mayo hasta Junio.
- Un modificador debe llevar valores y una barra inclinada (/), e indicar cuántas unidades se incrementan entre valores. Un valor de mes de */3 significa cada 3 meses. Un valor para mes de 4-9/2 significa los meses 4, 6 y 8.

La shell ejecuta los comandos, por lo que puede usar las funcionalidades descritas en este capítulo.

Algunos ejemplos que usan comandos directos en lugar de scripts son:

```
5 * * * * rm /tmp/*.gif # elimina todos los archivos GIF cada 5 minutos
5 * * * * rm -v /tmp/*.gif >> /tmp/gif.log # lo mismo, pero quedando
registrado en un log
```

Cuando cron ejecuta el comando, envía por correo electrónico la salida estándar y la salida de errores estándar al propietario del archivo crontab. Para evitar que el correo se sature con correos, puede redirigir la salida estándar y la salida de errores estándar a otro lugar:

```
command > /dev/null 2>&1
```

Problemas con los lenguajes de script

El principal uso de una shell es ejecutar comandos y trabajar con archivos, y las shells fueron diseñadas para hacer estas operaciones más fáciles. Otras tareas como realizar cálculos aritméticos son más duras, puesto que el texto necesita protegerse de la división de palabras y de la expansión mediante *. En shell scripts más complejos, la pila de paréntesis, corchetes y otros símbolos puede llegar a ser caótica.

En tiempos pasados (cuando teníamos ceros y unos, y podíamos sentirnos afortunados por tener unos) los artículos how-to a menudo proporcionaban grandes shell scripts a los usuarios: para descargar y compilar paquetes, hacer copias de seguridad, etc. Hoy en día, es preferible hacer estas tareas usando lenguajes de script más avanzados, por varias razones:

- Con el tiempo, aplicaciones como adduser y apt-get han automatizado algunas tareas tradicionales de los shell scripts.
- Los shell scripts no son fáciles ni de ampliar ni de mantener.
- Los shell scripts son más lentos.
- La sintaxis Shell induce a errores.

Perl inicialmente vino a llenar el hueco para los administradores que buscaban herramientas más productivas, pero ahora PHP le ha arrebatado el nicho Web, y Python ha ganado reputación y productividad. Escribiremos una aplicación en cada uno de estos lenguajes, algunos otros como ruby y Tcl también están disponibles en Linux.

Nuestra aplicación buscará en el archivo /etc/passwd el nombre, ID de usuario, talla o cualquier cosa que haya allí. Verá cómo abrir un archivo, leer registros, parsear formatos, buscar patrones e imprimir patrones. Luego buscaremos formas de evitar parte de este trabajo puesto que sudor != productividad. Será capaz de aplicar estas técnicas a otros archivos, como logs o páginas Web. Esto es un ejemplo de reutilización y seguramente ya lo haya hecho otras veces.

Inventémonos algunos requisitos para nuestras aplicaciones y expresémoslos en forma de pseudocódigo:

```
leer una cadena de búsqueda introducida por el usuario
abrir el archivo
por cada línea:
   parsear los campos (columnas)
   buscar la coincidencia entre los campos
   si hay alguna coincidencia
      imprimir los otros campos en un formato legible
```

En este momento, muchos programadores se habrían puesto ya a teclear (sin ni siquiera haber leído los requisitos o el formato de los datos). Los lectores de este libro son más disciplinados, además de más avispados. Han tenido que corregir los fallos que otros programadores han cometido y no quieren repetir estos mismos errores.

Formato de los datos: El archivo /etc/passwd

El archivo de contraseñas a menudo contiene cuentas estándar del sistema además de la cuenta de root, cuentas de aplicaciones como Apache y cuentas de usuarios. Aquí mostramos porciones de dicho archivo:

```
# System
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
...
# Applications
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
apache:x:48:48:Apache:/var/www:/bin/false
...
# Users
adedarc:x:500:500:Alfredo de Darc:/home/adedarc:/bin/bash
rduxover:x:501:501:Ransom Duxover:/home/rduxover:/bin/bash
cbarrel:x:502:502:Creighton Barrel:/home/cbarrel:/bin/bash
cmaharias:x:503:503:C Maharias:/home/cmaharias:/bin/bash
pgasquette:x:504:504:Papa Gasquette:/home/pgasquette:/bin/bash
bfrapples:x:505:505:Bob Frapples:/home/bfrapples:/bin/bash
```

Los campos separados por puntos son:

- Nombre de cuenta.
- Contraseña encriptada, o x si está usando /etc/shadow.
- User ID (uid).
- Group ID (gid).
- Nombre completo o descripción.
- Directorio personal.
- Shell.

Estamos interesados en el quinto campo (nombre completo o descripción). En los antiguos Unix, este campo se llamaba gecos, por razones que estaban obsoletas incluso entonces.

Versiones de script

Empezaremos cada una de las siguientes secciones con un script mínimo que busca una cadena en el archivo /etc/passwd e imprime la línea que coincida. Sabemos que esto es demasiado grande, pero queremos que lo que más nos interesa es que el script funcione.

Luego, dividiremos las líneas de entrada en campos y restringiremos la búsqueda de patrones al campo gecos que contiene nuestros nombres de usuario.

Luego, restringiremos la búsqueda a las líneas en que el valor del campo uid es mayor que 500. En nuestro caso, ID de usuario normales empiezan por el 501, por lo que excluiremos cuentas del sistema y de otros autómatas.

En este punto, estaremos ya cansados de los pasos previos, por lo que buscaremos algunas herramientas que puedan hacer este trabajo por nosotros.

El bash script

La mayoría de los lenguajes ofrecen librerías de funciones para varias tareas. Los programas desempeñan esta función para la shell, y los codificadores de shell scripts expertos están familiarizados con la mayoría de las utilidades Linux (cat, head, tail, awk, cut, grep, egrep y otros). Usaremos algunas de estas para nuestro bash script.

Aquí mostramos una versión rápida y poco legible (finduser.sh) que lee la cadena de búsqueda del usuario como parámetro, busca coincidencias en cualquier lugar de la línea e imprime cualquier línea que coincida:

```
#!/bin/bash
grep -i "$1" /etc/passwd
admin@server1:~$ chmod +x finduser.sh
admin@server1:~$ ./finduser.sh alf
adedarc:x:500:500:Alfredo de Darc:/home/adedarc:/bin/bash
```

Sería más rápido si solo tecleáramos:

```
Admin@server1:~$ grep -i alf /etc/passwd
```

¿Pero qué pasa si alf también coincide con una cuenta del sistema llamada gandalf?, ¿o una cadena en algún otro campo? Si queremos restringir la búsqueda al campo nombre para cuentas de usuarios normales (por ejemplo, cuentas con ID de usuario mayores que 500), nuestro script va a crecer un poquito.

Si profundizamos un poco en la documentación bash descubriremos que bash puede dividir la entrada en caracteres que no sean espacios, usando la variable IFS. En la siguiente versión del script, leemos el archivo /etc/passwd línea por línea, dividiendo cada línea en variables de campos. Si encontramos una coincidencia, necesitamos reconstruir la línea para imprimirla en su forma original:

```
#!/bin/bash
pattern=$1
IFS=":"
while read account password uid gid name directory shell
   do
   # Exact case-sensitive matches only!
   if [[ $name == $pattern ]]; then
      echo "$account:$password:$uid:$gid:$name:$directory:$shell"
```

```
      fi
      done < /etc/passwd
```

Pero ahora tenemos un problema con la búsqueda de coincidencias, al contrario que grep, bash no tiene una búsqueda parcial de cadenas que sea sensible a mayúsculas y a minúsculas. Tendremos que usar un patrón de búsqueda más sofisticado con ayuda externa, egrep:

```
#!/bin/bash
pattern=$1
IFS=":"
while read account password uid gid name directory shell
   do
   if [[ $(echo $name | egrep -i -c "$pattern") -gt 0 ]]; then
      echo "$account:$password:$uid:$gid:$name:$directory:$shell"
      fi
      done < /etc/passwd
```

Para nuestro script final, añadiremos una comprobación de los números uid:

```
#!/bin/bash
pattern=$1
IFS=":"
while read account password uid gid name directory shell
   do
   # Exact matches only!
   if [[ $uid -gt 500 && $(echo $name | egrep -i -c "$pattern")
   -gt 0 ]]; then
      echo "$account:$password:$uid:$gid:$name:$directory:$shell"
      fi
      done < /etc/passwd
```

Si ejecuta un shell script con una opción –v o –x, bash imprimirá cada comando antes de ejecutarlo. Esto le ayudará a ver lo que actualmente está haciendo el script.

El script Perl

Perl es conciso, y es realmente bueno trabajando con texto. Un equivalente para nuestro primer bash script es:

```
admin@server1:~$ perl -ne 'printf if 7alf/i' /etc/passwd
```

El /patron/ coincide con la cadena de búsqueda mientras que i ignora si son mayúsculas o minúsculas. Aquí mostramos un equivalente a la versión del script que usaremos como programa que cumple nuestros otros requisitos.

```
#!/usr/bin/perl
my $pattern = shift;
```



```
while (<>) {
   if (/$pattern/i) {
      print;
   }
}
```

Muchos elementos de la sintaxis de Perl son muy complicados, pero otros han sido heredados de la sintaxis de Shell (o de otras herramientas Unix comunes) y por tanto no son tan difíciles de recordar una vez que conoce esas herramientas. En particular, puede ver sentencias `while` e `if` en el script anterior, y se comportan de la misma forma en que lo harían los equivalentes Shell. La sintaxis <> es también una herencia de la < redirección > shell; provoca que cada iteración de un bucle `while` lea una línea de la entrada. Fíjese en que a diferencia de bash, las variables en Perl necesitan el $ inicial incluso cuando está asignando valores. La sentencia de impresión muestra lo que <> encuentra.

Perl tiene una forma alternativa de sintaxis que ahorra unos cuantos caracteres:

```
#!/usr/bin/perl
my $pattern = shift;
while (<>) {
   print if /$pattern/i;
}
```

El script (llamado `finduser.pl`) asume que el archivo de contraseñas se lee desde la entrada estándar, por lo que debería ejecutar algo como esto:

```
admin@server1:~$ ./finduser.pl alf < /etc/passwd
```

La siguiente versión abre el archivo de contraseñas directamente:

```
#!/usr/bin/perl
my $fname = "/etc/passwd";
my $pattern = shift;
open(FILE, $fname) or die("Can't open $fname\n");
while (<FILE>) {
   if (/$pattern/i) {
      print;
   }
}
close(FILE);
```

Para restringir las coincidencias al nombre del campo como hicimos en la sección bash, jugamos con algunas de las fortalezas de Perl:

```
#!/usr/bin/perl
my $fname = "/etc/passwd";
my $pattern = shift;
open(FILE, $fname) or die("Can't open $fname\n");
```

```
while (<FILE>) {
   $line = $_;
   @fields = split/:/;
   if ($fields[4] =~ /$pattern/i) {
      print $line;
   }
}
close(FILE);
```

El parámetro proporcionado por el usuario se lee en la variable $pattern usando la sentencia `shift`. El script también define otro tipo de variable: un vector llamado @fields. La función de división de Perl pone cada elemento separado por los dos puntos en una celda del vector. Podemos extraer el elemento número 4 (que realmente es el quinto elemento, porque los elementos se empiezan a numerar desde o) y compararlo sin atender a mayúsculas ni minúsculas con el parámetro proporcionado por el usuario.

Todos estos scripts han necesitado leer líneas de texto y buscar patrones, Debido a que `/etc/passwd` es un archivo muy importante en Linux, se le debería haber ocurrido pensar que alguien pudiera haber automatizado este trabajo. Afortunadamente, así ha sido: el viejo Perl ofrece una función llamada getpwent que devuelve el contenido de `/etc/passwd` proporcionando una línea cada vez que se llama y en la `forma` de un vector de cadenas. En la siguiente versión de nuestro script, asignamos a cada campo su propia variable; en la siguiente, usaremos el vector @list para capturarlos todos. En cada caso, queremos el campo gecos (llamado gcos en la documentación de Perl). Fíjese en que este es el campo 6 devuelto por getpwent, no el campo 4, porque getpwent soporta otros dos campos que aparecen en los archivos de contraseñas de sólo algunos sistemas:

```
#!/usr/bin/perl
$pattern = shift;
while (($name,$passwd,$uid,$gid,
      $quota,$comment,$gcos,$dir,
      $shell,$expire) = getpwent) {
   if ($gcos =~ /$pattern/i) {
      print "$gcos\n";
   }
}
```

```
#!/usr/bin/perl
$pattern = shift;
while (@fields = getpwent) {
   If ($fields[6] =~ /$pattern/i) {
      print "$fields[6]\n";
   }
}
```

Para flagelarnos un poco más, restrinjamos las búsquedas a los usuarios normales (uid > 500). Es una adición fácil.


```
#!/usr/bin/perl
$pattern = shift;
while (@fields = getpwent) {
   if ($fields[6] =~ /$pattern/i and $fields[2] > 500) {
      print "$fields[6]\n"
   }
}
```

El script PHP

PHP puede ser ejecutado por un servidor Web (usando CGI) o de manera autónoma (usando CLI). Nosotros usaremos la versión CLI. Si no tiene la versión CLI, puede instalarla en sistemas basados en Debian con apt-get install php4-cli.* Nuestro primer script PHP se parecerá mucho a los primeros script Perl:

```
#!/usr/bin/php
<?
$pattern = $argv[1];
$file = fopen("/etc/passwd", "r");
while ($line = fgets($file, 200)) {
   if (eregi($pattern, $line))
      echo $line;
}
fclose($file);
?>
```

Gracias a que en su origen era un complemento para páginas Web, PHP supone que el contenido por defecto del archivo debe ser interpretado como texto plano, y el código PHP se reconoce porque está comprendido entre símbolos <? o <?php y ?>. Muestra texto en la salida estándar. La función eregi hace una comparación de expresiones regulares sin tener en cuenta las mayúsculas y las minúsculas. Debido a que PHP ha tomado cosas prestadas de Perl, no es una sorpresa que tenga una función de división:

```
#!/usr/bin/php
<?
$pattern = $argv[1];
$file = fopen("/etc/passwd", "r");
while ($line = fgets($file, 200)) {
   $fields = split(":", $line);
   if (eregi($pattern, $fields[4]))
      echo $line;
}
fclose($file);
?>
```

Pero, ¿podemos invocar una función como getpwent de Perl para trocear el archivo de contraseñas para nosotros? PHP no parece tener un equivalente, por

lo que procederemos con parsep para restringir la búsqueda para los valores uid por encima de 500:

```
#!/usr/bin/php
<?
$pattern = $argv[1];
$file = fopen("/etc/passwd", "r");
while ($line = fgets($file, 200)) {
   $fields = split(":", $line);
   if (eregi($pattern, $fields[4]) and $fields[2] > 500)
      echo $line;
}
fclose($file);
?>
```

El script Python

Los script Python son distintos de los de Perl y de los de PHP, porque las sentencias terminan con espacios en blanco en lugar de con puntos y coma o llaves al estilo C. Todas las tabulaciones también tienen sentido. Nuestro primer script Phyton, al igual que nuestros primeros intentos en los otros lenguajes, busca el archivo de contraseñas e imprime cualquier línea que contenga el texto coincidente:

```
#!/usr/bin/python
import re, sys
pattern = "(?i)" + sys.argv[1]
file = open("/etc/passwd")
for line in file:
   if re.search(pattern, line):
      print line
```

Python tiene espacios de nombres (Perl también) para agrupar funciones, que es por lo que las funciones del script están precedidas por cadenas del tipo sys. y re.. Esto ayuda a mantener la modularidad del código. La "(¿i)" en la tercera línea del código del script hace que la búsqueda no distinga entre mayúsculas y minúsculas, similar a la /i en Perl.

La siguiente iteración divide la línea de entrada en campos:

```
#!/usr/bin/python
import re, sys
pattern = "(?i)" + sys.argv[1]
file = open("/etc/passwd")
for line in file:
   fields = line.split(":")
   if re.search(pattern, fields[4]):
      print line
```


Python tiene un equivalente a la función getpwent de Perl que permite restringir la búsqueda a los campos que contienen nombres. Guarde el siguiente script como `finduser.py`:

```
#!/usr/bin/python
import re, sys, pwd
pattern = "(?i)" + sys.argv[1]
for line in pwd.getpwall( ):
   if re.search(pattern, line.pw_gecos):
      print line
```

Ahora veamos cómo trabaja:

```
admin@server1:~$ ./finduser.py alf
('adedarc', 'x', 501, 501, 'Alfredo de Darc', '/home/adedarc', '/bin/bash')
```

En este script, la línea que hemos imprimido es una lista Python en lugar de una cadena, de esta forma queda mejor pintada. Para imprimir la línea en su formato original, use esto:

```
#!/usr/bin/python
import re, sys, pwd
pattern = "(?i)" + sys.argv[1]
for line in pwd.getpwall( ):
   if re.search(pattern line.pw_gecos):
      print ":".join(["%s" % v for v in line])
```

La última línea es necesaria para convertir cada campo en una cadena (pw_uid y pw_gid son enteros) antes de juntarlos en una cadena larga separada por (:). Aunque Perl y PHP le permiten tratar una variable como una cadena o un número, Python es más estricto.

El paso final es restringir todas las búsquedas para que busque en cuentas con uid > 500:

```
#!/usr/bin/python
import re, sys, pwd
pattern = "(?i)" + sys.argv[1]
for line in pwd.getpwall( ):
   if line.pw_uid > 500 and re.search(pattern line.pw_gecos):
      print ":".join(["%s" % v for v in line])
```

Escogiendo un lenguaje de script

La elección de un lenguaje de programación, al igual que la elección de un editor de texto o un sistema operativo, depende del gusto. Algunas personas creen que Perl es ilegible, y otras se resisten a usar las reglas de espacios de Python. A

menudo la comparación no suele ir más lejos; si no le gusta la remolacha, por qué comerla.

Si está a gusto con el estilo del lenguaje, el criterio más importante es la productividad de la tarea. bash es una `forma` rápida de crear script cortos, incluso de una línea, pero se hace más pesado cuando son cientos de líneas. Perl puede ser `di`fícil de leer, pero es muy potente y se beneficia de la gran librería CPAN. PHP se parece a C, carece de espacios de nombres, el código y la salida es amigable, y tiene algunas librerías muy buenas. Python es quizás el más fácil de leer y escribir, lo que supone una ventaja especial para scripts grandes.

Otras lecturas

El apéndice contiene algunos scripts más grandes que pueden ser útiles para administradores de sistemas. "Linux Shell Scripting with Bash de Ken Burtch (Sams)" y "Advanced Bash- Scripting Guide" (`http://www.tldp.org/LDP/abs/html`) son buenos recursos. Si se quiere aventurar con otros lenguajes de script, cualquier libro con un animal en la carátula será una apuesta segura (a menos que quiera buscar COBOL en la sección de niños).

Capítulo 11

Haciendo copia de seguridad de los datos



Los ordenadores fallan, los discos se rompen, los chips se funden, los cables tienen cortocircuitos y la bebida cae sobre los aparatos. Algunos ordenadores se roban o son víctimas de errores humanos. Puede perder ya no sólo el hardware y el software, sino lo que es más importante, los datos. Restaurar los datos perdidos lleva tiempo y dinero.

Mientras tanto, sus clientes no estarán contentos, y el gobierno investigará si los datos que almacena cumplen con la legislación. Hacer copias de seguridad de todos los datos importantes es un seguro barato contra potenciales desastres que pueden salir muy caros, y la continuidad comercial requiere un plan de salvaguarda y restauración.

En este capítulo cubriremos varias herramientas para hacer copias de seguridad, pues puede ser útil en determinadas circunstancias:

- rsync: Suficiente para la mayoría de los archivos de usuario; transfiere archivos eficientemente por una red a otro sistema, desde el cual puede recuperarlos en caso de que le ocurra un desastre al sistema local.
- tar: Programa tradicional Unix para crear colecciones comprimidas de archivos; crea convenientes bloques de datos que puede salvaguardar usando otras herramientas de este capítulo.
- cdrecord/cdrtools: Graba archivos en CD o DVD.
- Amanda: Automatiza las copias en cinta, útil en entornos con grandes cantidades de datos.
- MySQL tools: Ofrece formas para solucionar los requisitos particulares de las bases de datos.

Salvaguardando los datos de usuario en un servidor con rsync

Los datos más críticos para salvaguardar son los datos que es imposible, o muy costoso, recomponer. Normalmente suelen ser datos de usuario que han ido creciendo durante meses o años de trabajo. Normalmente puede restaurar los datos del sistema de manera relativamente fácil reinstalando la distribución original.

Nos centraremos en hacer copias de seguridad de datos de usuario de equipos Linux de escritorio. Un servidor de copias de seguridad necesita suficiente espacio en disco para almacenar todos los archivos de usuario. Se recomienda un equipo dedicado. Para una oficina grande, los discos deberían tener una configuración en RAID (*Redundant Array of Independent Disks*) para protegerse contra fallos.

La utilidad Linux rsync es una copia de un programa diseñado para replicar grandes cantidades de datos. Puede saltarse archivos copiados previamente y fragmentos y encriptar las transferencias de datos con ssh, haciendo copias de seguridad remota con rsync de manera más rápida y más segura que con herramientas tradicionales como cp, cpio o tar. Para comprobar si rsync está en su sistema, introduzca:

```
# rsync --help
bash: rsync: command not found
```

Si ve este mensaje, tendrá que conseguir el paquete rsync. Para instalarlo en Debian, introduzca:

```
# apt-get install rsync
```

Normalmente, hará copias de seguridad para preservar los permisos originales. Es decir, necesitará asegurarse de que todos los usuarios tienen cuentas y directorios personales en el servidor de copias de seguridad.

Aspectos básicos de rsync

La sintaxis del comando rsync es:

```
rsync opciones origen destino
```

Algunas de las principales opciones de línea de comandos para rsync son las siguientes:

- -a: Archivo. Esta opción cumple la mayoría de los requisitos mencionados previamente, y es más fácil de teclear y de pronunciar que su equivalente, -Dgloprt.
- -b: Hace copias de seguridad de todos los archivos destino en lugar de sustituirlos. Normalmente no tendrá que usar esta opción, a menos que quiera mantener las versiones antiguas de cada archivo. Puede provocar que el servidor de copias de seguridad se sature rápidamente.
- -D: Esta opción se usa cuando se replican sistemas de archivos; no es necesario para archivos de usuario. Funciona sólo cuando rsync se ejecuta como root. Está incluida en –a.
- -g: Preserva los permisos de grupo de los archivos que se están duplicando. Es importante para las copias de seguridad. Está incluida en –a.
- -H: Preserva los enlaces. Si dos nombres han sido replicados refiriéndose al mismo inodo de archivo, esto mantiene la misma relación en el destino. Esta opción hace que rsync vaya más lento, pero se recomienda su uso.
- -l: Copia enlaces simbólicos como enlaces simbólicos. Casi siempre tendrá que incluir esta opción; sin ella, un enlace simbólico podría copiarse como un archivo normal. Está incluida en –a.
- -n: Ejecución seca: muestra los archivos que se transferirán, pero no muestra los que se están transfiriendo.
- -o: Preserva la autoría de usuario de los archivos que se están replicando. Es importante para las copias de seguridad. Está incluida en –a.
- -p: Preserva los permisos de archivos. Es importante para las copias de seguridad. Está incluida en –a.
- -P: Activa --partial y --progress.
- --partial: Activa las transferencias de archivos parciales. Si rsync se para, será capaz de completar el resto del archivo cuando rsync vuelva a iniciarse.
- --progress: Muestra el progreso en la transferencia de archivos.
- -r: Activa la recursividad, transfiriendo todos los subdirectorios. Incluido en –a.
- --rsh='ssh': Usa SSH para transferencia de archivos. Se recomienda su uso debido a que el protocolo de transferencia por defecto (rsh) no es seguro. También puede definir la variable de entorno RSYNC_RSH como ssh para obtener el mismo efecto.

- -t: Preserva la hora de modificación de cada archivo. Incluido en –a.
- -v: Lista los archivos que se están transfiriendo.
- -vv: Igual que -v, pero también lista los archivos que se ignoran.
- -vvv: Igual que –vv, pero también imprime información de depuración de rsync.
- -z: Activa la compresión, más útil sobre Internet que sobre una LAN de alta velocidad.

Hay muchas más opciones para rsync que pueden ser útiles en situaciones concretas. Podrá encontrarlas en las páginas-manual.

Después de las opciones, vienen los parámetros origen y destino. Tanto uno como otro pueden ser rutas locales del ordenador donde rsync se está ejecutando, las designación de un servidor (normalmente usado para descargar servidores de archivos) o designaciones del tipo user@host:path para ssh. Debido a que rsync toma muchas opciones y argumentos muy largos que no cambian regularmente, luego escribiremos un bash script para ejecutarlo.

Haciendo un script para copias de seguridad de usuario

Esta sección presenta un bash script sencillo que hace una copia de seguridad desde el escritorio de un usuario al servidor de copias de seguridad. El nombre del servidor de copias de seguridad está asignado a la variable dest en el script. La variable user está asignada al nombre de usuario de la cuenta que ejecuta el script, puesto que invoca el comando whoami y captura la salida como cadena.

El comando cd cambia el directorio actual al directorio personal del usuario. La condición lógica OR que sigue al comando cd aborta el script si hay algún fallo. El punto (.) especifica que el directorio actual es el parámetro origen.

Para el parámetro destino, especificamos el nombre de usuario y el nombre de equipo para autentificarse vía ssh, seguido de un punto para especificar que el directorio personal actual es el destino en el equipo.

Este es el script:

```
#!/bin/bash
export RSYNC_RSH=/usr/bin/ssh
dest=backup1
user=$(whoami)
cd || exit 1
rsync -aHPvz . ${user}@${dest}:.
```


La variable de entorno RSYNC_RSH contiene el nombre de la shell que rsync usará. Por defecto es `/usr/bin/rsh`, por lo que la cambiaremos por `/usr/bin/ssh`.

Ejecutar este script replica todos los archivos del directorio personal del usuario que lo ejecuta desde su directorio personal en el servidor de copias de seguridad.

Veamos cómo funciona para un usuario de ejemplo (después de haberse autentificado en su escritorio):

```
amy@desk12:~$ ./backup
Password:
building file list ...
14 files to consider
./
new-brochure.sxw
      37412 100%  503.91kB/s     0:00:00  (1, 62.5% of 16)
sales-plan-2006-08.sxw
      59513 100%  1.46MB/s       0:00:00  (2, 68.8% of 16)
sales-plan-2006-09.sxw
      43900 100%  691.47kB/s     0:00:00  (3, 75.0% of 16)
sales-plan-2006-10.sxw
      41285 100%  453.00kB/s     0:00:00  (4, 81.2% of 16)
vacation-request.sxw
      15198 100%  154.60kB/s     0:00:00  (5, 87.5% of 16)
sent 185942 bytes received 136 bytes 24810.40 bytes/sec
total size is 210691 speedup is 1.13
amy@desk12:~$
```

rsync nos indica que está considerando 14 archivos. Hace copia de seguridad sólo de 5, puesto que los otros nueve ya están en el servidor de copias de seguridad y no han cambiado.

Esta salida muestra que el progreso es del 100 por 100 cuando todos los archivos están completos, e indica cuánto tardará cada transferencia. En una LAN de alta velocidad el tiempo de transferencia será menos de un segundo para archivos de tamaño pequeño o medio.

En conexiones más lentas o de archivos muy grandes, verá el estado del progreso que proporciona el tamaño y el porcentaje transferido y una estimación de tiempo que falta hasta completarse.

Listando archivos en el servidor de copias de seguridad

rsync puede ofrecer una lista de los archivos que hay en el servidor de copias de seguridad. Es útil para verificar cuando hay archivos nuevos o importantes, así como para encontrar archivos que necesitan restaurarse debido a

que han estado perdidos o debido a que el usuario necesita recuperar una versión antigua.

Para obtener este listado, omita las opciones y los parámetros de destino. Aquí mostramos un bash script sencillo que obtiene los resultados deseados:

```
#!/bin/bash
dest=server1
user=$(whoami)
cd || exit 1
rsync "${user}@${dest}:." | more
```

Ejecutar este script produce resultados parecidos al siguiente:

```
amy@desk12:~$ ./backlist
Password:
drwx------   4096  2006/08/09  13:20:41  .
-rw-------  10071  2006/08/09  12:35:21  .bash_history
-rw-r--r--    632  2006/07/27  23:03:06  .bash_profile
-rw-r--r--   1834  2006/07/26  19:59:08  .bashrc
-rwxr-xr-x    108  2006/07/27  23:06:51  .path
-rwxr-xr-x     79  2006/08/09  13:18:34  backlist
-rwxr-xr-x    137  2006/08/09  13:19:29  backrestore
-rwxr-xr-x     88  2006/08/09  13:03:46  backup
-rw-r--r--  37412  2006/07/17  14:40:52  new-brochure.sxw
-rw-r--r--  59513  2006/07/19  09:16:41  sales-plan-2006-08.sxw
-rw-r--r--  43900  2006/07/19  22:51:54  sales-plan-2006-09.sxw
-rw-r--r--  41285  2006/07/17  16:24:19  sales-plan-2006-10.sxw
-rw-r--r--  15198  2006/07/10  14:42:23  vacation-request.sxw
drwx------   4096  2006/08/09  13:12:25  .ssh
amy@desk12:~$
```

Restaurando archivos perdidos o dañados

Ningún sistema de copias de seguridad es bueno si los archivos perdidos no pueden restaurarse. No sólo debe estar listo para el caso de que se produzca un desastre, sino que también debe comprobar que los planes de recuperación y restauración funcionarán en caso de necesitarlos.

Nuestro script de restauración es solamente un poco más complicado que el script previo. Hemos añadido una forma de especificar los archivos individuales que deben restaurarse:

```
#!/bin/bash
dest=server1
user=$(whoami)
cd || exit 1
for file in "$@" ; do
   rsync -aHPvz "${user}@${dest}:./${file}" "./${file}"
done
```


Para restaurar archivos, simplemente ejecutamos el script, le pasamos los nombres de los archivos que se van a restaurar como parámetros de línea de comandos.

En el siguiente ejemplo, eliminaremos intencionadamente uno de nuestros archivos y luego veremos cómo se restaura:

```
amy@desk12:~$ rm sales-plan-2006-10.sxw
amy@desk12:~$ ./backrestore sales-plan-2006-10.sxw
Password:
receiving file list ...
1 file to consider
sales-plan-2006-10.sxw
   41285 100%   6.56MB/s    0:00:00  (1, 100.0% of 1)

sent 42 bytes  received 39299 bytes  6052.46 bytes/sec
total size is 41285  speedup is 1.05
amy@desk12:~$
```

También podemos restaurar todos los archivos a la vez usando un punto como nombre de archivo.

Copias de seguridad automatizadas

Las copias de seguridad pueden automatizarse usando script similares a estos ejecutados como tareas cron (descritas en el capítulo anterior). SSH requiere que se introduzca la contraseña de usuario, por lo que necesitará incluir las claves públicas de sus usuarios en la configuración SSH para hacer que la autentificación remota funcione cuando los usuarios no están presentes (por ejemplo, a las 3:00 a.m.).

Tiene muchas opciones para crear copias de seguridad. Puede ejecutar un script de tarea cron en el servidor diaria o semanalmente, para hacer copias de seguridad en otro servidor. Los negocios con oficinas remotas necesitarán hacer copias regulares de los datos de dichas oficinas por Internet. Las copias también se puede volcar a un CD o un DVD, a una cinta y después archivarlas y transportarlas a un lugar seguro.

Archivos tar

El comando tar crea un archivo a partir de uno o más archivos o directorios. También puede listar todo el contenido de un archivo, o extraer varios archivos y directorios de un archivo. A los archivos tar también se les suele llamar tarfile o tarball.

Un archivo tar ofrece algunas ventajas sobre un directorio con archivos separados. Por ejemplo, es más fácil enviar un directorio completo a través de correo electrónico. Los directorios contienen muchos archivos similares que pueden comprimirse más eficientemente cuando la compresión opera en todos los datos de un único archivo.

Un uso común para un archive tar es ayudar en la distribución del código fuente del software libre o de código abierto. En la mayoría de los casos, los archivos tar están comprimidos con programas como gzip o bzip2. Sin embargo, si todos los archivos que se están almacenando están comprimidos (lo que es cierto en archivos de audio, video y Open Office), comprimir el archivo no producirá mucho beneficio.

Puede llamar a un archivo tar como quiera, pero se usan ciertas extensiones de archivo de manera convencional para indicarle a los receptores cómo descomprimir el archivo.

Las extensiones más comunes son:

- .tar: Para archivos tar descomprimidos.
- .tar.gz or .tgz: Para archivos tar que han sido comprimidos con el programa de compresión gzip.
- .tar.bz2 or .tbz: Para archivos tar que han sido comprimidos con el programa bzip2.

La sintaxis del comando tar es:

```
tar opciones argumentos
```

Las opciones no vienen precedidas por el tradicional guión (-), aunque muchas versiones de tar si aceptan el guión.

Las opciones más útiles son:

- -b: Especifica el tamaño de bloque (por defecto son 512 bytes).
- -c: Crea (escribe) un nuevo archivo.
- -f filename: Lee desde o escribe en un archivo. Si el nombre del archivo se omite o es -, el archivo se escribirá en la salida estándar o se leerá de la entrada estándar.
- -j: Comprime o descomprime el archivo usando bzip2 o bunzip2. Los archivos comprimidos con bzip2 normalmente tienen el sufijo .bz2.
- -p: Preserva los permisos de archivo.
- -t: Lista los archivos en un archive existente.

- -v: Al crear o desempaquetar archivos, lista el contenido. Con la opción -t se ofrecen más detalles acerca de los archivos listados.
- -x: Extrae (lee) archivos desde un archivo existente.
- -z: Comprime o descomprime el archivo usando gzip o gunzip. Los archivos comprimidos con gzip normalmente tienen el sufijo .gz.

Creando un nuevo archivo

Puede crear un archive tar sólo para guardar el grupo de archivos para sus propios propósitos, para enviarlos por correo electrónico, o para hacerlos públicos (por ejemplo, en un servidor FTP). Algunos comandos típicos para archivar los documentos de trabajo de un directorio son:

- Para crear el archivo work-docs.tar a partir de los distintos documentos de un directorio:

```
$ tar -cf work-docs.tar work-docs
```

- Para crear el archivo comprimido work-docs.tar.gz a partir de los documentos de un directorio:

```
$ tar -czf work-docs.tar.gz work-docs
```

- Para crear el archivo comprimido work-docs.tar.bz2 a partir de los documentos de un directorio:

```
$ tar -cjf work-docs.tar.bz2 work-docs
```

Extrayendo datos de un archivo

Algunas veces, puede necesitar extraer archivo de un archivo que ya ha creado (como una copia de seguridad), a partir de un archivo que alguien le ha enviado, o a partir de un archivo que se ha descargado de Internet (por ejemplo, el código fuente de algún tipo de software que necesita).

Antes de extraer un archivo, debería listar y revisar todo el contenido. No querrá sobrescribir accidentalmente los archivos existentes en su sistema con los archivos del archivo, ni volver a tener archivos que había borrado previamente.

Los archivos de un archivo deberían estar organizados en un directorio, pero casi nadie lo hace, por lo que tendrá que tener cuidado para evitar extraer archivos en su directorio actual. Normalmente, es una buena idea crear un nuevo directorio en su equipo en el que extraer el archivo tar. Esto mantiene los archi-

vos extraídos separados de los otros archivos, por lo que no llegarán a mezclarse. De esta forma se evita que la extracción sobrescriba archivos existentes.

La opción –t lista los nombres de los archivos y de los directorios que se crearán al descomprimir el archivo. Añadir la opción –v incrementa la aparición de mensajes que ofrecen detalles acerca de cada archivo del archivo tar, incluyendo el tamaño de cada archivos y su última fecha de modificación. Aquí mostramos algunos ejemplos de comandos:

- Para listar los archivos del archivo collection.tar:

```
$ tar -tf collection.tar
```

- Para listar los archivos del archivo collection.tar.bz2 con detalles extra:

```
$ tar -tvjf collection.tar.bz2
```

- Para extraer los archivos de collection.tar en el directorio actual, conservando los permisos originales:

```
$ tar -xpf collection.tar
```

  La opción –x extrae los archivos en el directorio actual. tar funciona de manera silenciosa a menos que se especifique la opción –v que se usa para listar los archivos. La opción –p mantiene los permisos originales, por lo que los archivos extraídos tendrán los mismos permisos que los archivos que fueron archivados.

- Para extraer los archivos de collection.tar.gz en el directorio actual manteniendo los permisos originales:

```
$ tar -xpzf collection.tar.gz
```

- Para extraer los archivos de collection.tar.bz2 en el directorio actual manteniendo los permisos originales:

```
$ tar -xpjf collection.tar.bz2
```

- Para listar y extraer los archivos de collection.tar.bz2 en el directorio actual manteniendo los permisos originales:

```
$ tar -xpvjf collection.tar.bz2
```

Un ejemplo completo de compresión y descompresión con tar

La siguiente sesión shell muestra la creación de un archivo tar a partir de un directorio de archivos:

```
amy@desk12:~$ ls -dl monthly-reports
drwxr-xr-x 2 amy amy 4096 2006-08-11 14:15 monthly-reports
```



```
amy@desk12:~$ ls -l monthly-reports
total 228
-rw-r--r-- 1 amy amy 50552 2006-05-09 11:09 mr-2006-04.sxw
-rw-r--r-- 1 amy amy 51284 2006-06-06 15:44 mr-2006-05.sxw
-rw-r--r-- 1 amy amy 51428 2006-07-06 14:30 mr-2006-06.sxw
-rw-r--r-- 1 amy amy 54667 2006-08-07 10:06 mr-2006-07.sxw
amy@desk12:~$ tar -czf monthly-reports-aug.tar.gz monthly-reports
amy@desk12:~$ ls -l monthly-reports-aug.tar.gz
-rw-r--r-- 1 amy amy 199015 2006-08-14 12:46 monthly-reports-aug.tar.gz
```

La siguiente sesión shell muestra el listado del contenido del archivo tar:

```
amy@desk12:~$ ls -l monthly-reports-aug.tar.gz
-rw-r--r-- 1 amy amy 199015 2006-08-14 12:46 monthly-reports-aug.tar.gz
amy@desk12:~$ tar -tzf monthly-reports-aug.tar.gz
monthly-reports/
monthly-reports/mr-2006-04.sxw
monthly-reports/mr-2006-05.sxw
monthly-reports/mr-2006-06.sxw
monthly-reports/mr-2006-07.sxw
amy@desk12:~$ tar -tvzf monthly-reports-aug.tar.gz
drwxr-xr-x amy/amy      0  2006-08-11 14:15:12 monthly-reports/
-rw-r--r-- amy/amy  50552  2006-05-09 11:09:12 monthly-reports/mr-
                                               2006-04.sxw
-rw-r--r-- amy/amy  51284  2006-06-06 15:44:33 monthly-reports/mr-
                                               2006-05.sxw
-rw-r--r-- amy/amy  51428  2006-07-06 14:30:19 monthly-reports/mr-
                                               2006-06.sxw
-rw-r--r-- amy/amy  54667  2006-08-07 10:06:57 monthly-reports/mr-
                                               2006-07.sxw
amy@desk12:~$
```

La siguiente sesión shell muestra la extracción del contenido del archivo tar:

```
amy@desk12:~$ mkdir extract.dir
amy@desk12:~$ cd extract.dir
amy@desk12:~/extract.dir$ tar -xzf ../monthly-reports-aug.tar.gz
amy@desk12:~/extract.dir$ tar -xvzf ../monthly-reports-aug.tar.gz
monthly-reports/
monthly-reports/mr-2006-04.sxw
monthly-reports/mr-2006-05.sxw
monthly-reports/mr-2006-06.sxw
monthly-reports/mr-2006-07.sxw
amy@desk12:~/extract.dir$ tar -xvvzf ../monthly-reports-aug.tar.gz
drwxr-xr-x amy/amy      0  2006-08-11 14:15:12 monthly-reports/
-rw-r--r-- amy/amy  50552  2006-05-09 11:09:12 monthly-reports/mr-
                                               2006-04.sxw
-rw-r--r-- amy/amy  51284  2006-06-06 15:44:33 monthly-reports/mr-
                                               2006-05.sxw
-rw-r--r-- amy/amy  51428  2006-07-06 14:30:19 monthly-reports/mr-
                                               2006-06.sxw
```

```
-rw-r--r-- amy/amy 54667  2006-08-07 10:06:57 monthly-reports/mr-
                                              2006-07.sxw
amy@desk12:~/extract.dir$ cd
amy@desk12:~$
```

Resumen

Las cosas más importantes de recordar sobre tar son:

- -c lee sus archivos y crea (escribe) un archivo tar.
- -x extrae (lee) un archivo tar y escribe sus archivos.

La mayoría de los administradores Unix y Linux han combinando estas opciones alguna vez.

Guardando archivos en medios ópticos

Los CD y DVD grabables, llamados CD-R, DVD-R y DVD+R, le permite guardar archivos de una forma compacta y conveniente. Pueden usarse para hacer copias de seguridad que pueden almacenar un sitio, para distribuir software o datos a los usuarios o a los clientes. Un CD-R puede almacenar hasta 700 MB de datos, mientras que un DVD-R o DVD+R pueden almacenar 4.7 GB. También existe una versión de capa dual para DVD+R, con una capacidad de 8.55 GB.

La diferencia entre un DVD-R y un DVD+R es la tecnología usada para ubicar el láser en la pista de grabado. Los dos métodos son incompatibles, por lo que si su unidad soporta o sólo DVD-R o sólo DVD+R, deberá usar el medio adecuado (existen unidades que soportan ambos formatos, permitiendo usar cualquiera de los dos).

Grabar archivos en un CD o DVD no es tan sencillo ni tan flexible como grabarlos en un disco duro. Los medios regrabables pueden saltarse algunas limitaciones, pero tienen un coste mayor y una compatibilidad más reducida. En esta sección, nos centraremos en guardar archivos en CD-R. Los métodos para los DVD son similares.

Un CD de datos consiste en un vector de sectores de 2048 bytes cada uno. Un sistema de archivos especial conocido como ISO-9660 que se usa para organizar los archivos en el CD por lo que puede leerse en un amplio rango de computadores y otros dispositivos. Los reproductores de CD de música más modernos también soportan CD de datos escritos en el formato ISO-9660, por lo que pueden acceder a los archivos de música en formatos comprimidos como MP3. Los DVD usan un sistema de archivos más moderno llamado *Universal Disk Format* (UDF).

Para grabar datos, todas las grabadoras de CD y la mayoría de las de DVD necesitan que haya un flujo continuo de datos hacia la unidad. Si los datos no

están disponibles cuando el láser se dispone a grabarlos, el láser se parará y se romperá así la continuidad del grabado. Los métodos usados para grabar CD fueron diseñados para sistemas informáticos más lentos, para maximizar la fiabilidad de estas grabaciones. Los ordenadores actuales asumen el reto de conseguir que no falten datos que grabar y a la vez ser más rápidos; no obstante muchas grabadoras disponen de mecanismos contra la insuficiencia de datos en el buffer. Los archivos que se grabarán se suelen agrupar en un archivo llamado archivo de imagen ISO, que normalmente tiene la extensión .iso. Este archivo se graba posterior y directamente sobre el CD-R. Es posible grabar directamente sobre el CD-R sin crear un archivo .iso primero, pero este método incrementa el riesgo de que ocurra algo en el computador y el proceso falle.

El software necesario para grabar un CD o un DVD en Linux, está ubicado en un paquete llamado cdrecord (fíjese en que este paquete puede formar parte de otro llamado cdrtools). Si este paquete todavía no está instalado en su sistema, debería instalarlo usando los métodos que ya ha aprendido. En Debian Sarge, debería ejecutar el comando:

```
# apt-get install cdrecord mkisofs
```

Debian 4.0 sustituyó el paquete cdrecord por otro llamado wodim. Otros paquetes incluyen dvd+rw-tools (descrito en at `http://www.debianhelp.co.uk/burningdvd.htm`) y K3b (`http://www.k3b.org`).

Accediendo a su unidad CD-R

Linux soporta el grabado en unidades IDE ATAPI CD-R a través del driver llamado ide-scsi. La mayoría de las distribuciones Linux también incluyen este driver en el núcleo. Si su sistema no tiene el driver, necesitará cargar el módulo del driver (instalándolo si fuese necesario) o posiblemente recompilar el núcleo.

El driver ide-scsi emula un dispositivo SCSI por software que está diseñado sólo para dispositivos SCSI. Su unidad IDE ATAPI CD y su unidad de DVD deberán aparecer como si fueran dispositivos SCSI cuando el driver ide-scsi esté activo.

El siguiente comando mostrará los dispositivos SCSI de su sistema, por lo que podrá ubicar el número de driver emulador de SCSI de su unidad CD-R. También se listarán otros dispositivos, incluyendo dispositivos SCSI reales si su computador realmente los tiene. Ejecute este comando como root:

```
# cdrecord -scanbus
```

La salida será similar a esta:

```
Cdrecord-Clone 2.01 (i686-pc-linux-gnu) Copyright (C) 1995-2004
J&#246;rg Schilling
```

```
scsidev: 'ATA'
devname: 'ATA'
scsibus: -2 target: -2 lun: -2
Linux sg driver version: 3.5.27
Using libscg version 'schily-0.8'.
scsibus1:
    1,0,0   100) 'SONY    ' 'CD-RW CRX195E1  ' 'ZYS5' Removable CD-ROM
    1,1,0   101) 'DVD-16X ' 'DVD-ROM BDV316E ' '0052' Removable CD-ROM
    1,2,0   102) *
    1,3,0   103) *
    1,4,0   104) *
    1,5,0   105) *
    1,6,0   106) *
    1,7,0   107) *
```

Busque la descripción del dispositivo que coincide con su grabador CD-R. Si tiene más de un dispositivo, el nombre de la marca y el modelo debería ayudarle a identificar el dispositivo correcto. La salida debería mostrar al menos CD-R o CD-RW en la descripción. En este ejemplo, nuestra grabadora de CS está en dispositivo SCSI emulado 1,0,0.

Si el driver ide-scsi no está instalado o no está activo, puede obtener una salida como esta:

```
Cdrecord-Clone 2.01 (i686-pc-linux-gnu) Copyright (C) 1995-2004
J&#246;rg Schilling
cdrecord: No such file or directory. Cannot open '/dev/pg*'. Cannot
open SCSI driver.
cdrecord: For possible targets try 'cdrecord -scanbus'.
cdrecord: For possible transport specifiers try 'cdrecord dev=help'.
cdrecord:
cdrecord: For more information, install the cdrtools-doc
cdrecord: package and read /usr/share/doc/cdrecord/README.ATAPI.setup.
```

Si obtiene este tipo de salida, necesitará activar el driver ide-scsi antes de pasar a la grabación.

Opciones por defecto

Pueden configurarse varios parámetros cdrecord. Por ejemplo, puede configurar cdrecord para reconocer los nombres de los dispositivos de grabado (por lo que no tiene que memorizar los números de dispositivo) y puede designar un dispositivo por defecto.

Para configurar cdrecord, autentifíquese como (o use su – para cambiar a) root. Luego cree el archivo de texto con su editor:

```
# vi /etc/default/cdrecord
```


Pondremos las siguientes líneas de texto en este archivo para que coincidan con los dispositivos de la salida anterior generada por cdrecord-scanbus. Necesitará cambiar estos valores para que coincidan con los de sus propios dispositivos. Elija algunos nombres para poner en lugar de cd y dvd. Los espacios en blanco entre campos deben ser tabulaciones, no espacios:

```
CDR_DEVICE=cd
cd=1,0,0        -1    -1      ""
dvd=1,1,0       -1    -1      ""
```

Si la versión de su núcleo es la 2.6, necesitará especificar el dispositivo con el prefijo ATA; debido al rediseño del driver. En este caso, el archivo de configuración debería quedar:

```
CDR_DEVICE=cd
cd=ATA:1,0,0    -1    -1      ""
dvd=ATA:1,1,0   -1    -1      ""
```

También puede definir la velocidad de grabado por defecto de cada unidad, a la derecha después del nombre del dispositivo. -1 indica que se usará el valor por defecto.

El siguiente número es el tamaño de buffer FIFO, una vez más, -1 especifica el valor por defecto del sistema Linux. El último elemento de la línea permite que le pase una opción específica del driver; la hemos dejado como una cadena vacía.

Las nuevas versiones del soporte cdrecord soportan la opción driveropts= burnfree para protegerse contra los desbordamientos de buffer.

Preparando los archivos para grabar un CD-R

El comando mkisofs crea un archivo de imagen ISO. Debería contener todos los archivos que se grabarán en el CD-R. Hay muchas opciones para este comando, pero las más importantes que usaremos son:

- -J: Incluye nombres Joliet para compatibilidad con Windows.
- -r: Incluye nombres Rock Ridge para compatibilidad Unix/Linux.
- -v: Activa el modo de palabra completa que muestra el estado del progreso.
- -V id_string: Especifica un ID de volumen para el disco que se creará.
- -o filename: Especifica el nombre de archivo de la imagen ISO que se está creando.

Aquí mostramos un comando de ejemplo para incluir todos los archivos de un directorio específico:

```
# mkisofs -JrvV "disc name" -o backup.iso /home/amy
```

Podrá ver una salida bastante grande para este comando. La salida es útil para archivos de colecciones grandes porque ayuda a estimar el tiempo que queda. Si prefiere no tener esta salida, omita la opción –v del comando.

Grabando el CD-R

Puede grabar un CD-R con la imagen ISO que ha creado. Para realizar la grabación actual, autentifíquese como (o use su- para cambiar a) root. Los permisos de root son necesarios para que el programa cdrecord pueda acceder a la capa SCSI, para modificar las prioridades del proceso y para bloquear el espacio en RAM y así evitar equivocaciones. La escritura de CD tiene dependencias temporales que son críticas, por tanto ayuda a mantener la actividad del sistema tan plana como sea posible. Si está usando un CD-RW en una unidad CD-RW, necesitará borrar el CD-RW antes de efectuar el grabado:

```
# cdrecord blank=fast padsize=63s -pad -dao -v -eject
```

Nota: Algunas unidades necesitan que el CD sea retirado para reiniciar la unidad y dejar la lista para la siguiente operación. A menos que haya descubierto que su unidad no necesita eso, use la ejecución –eject como se muestra aquí.

Para grabar la imagen ISO creada en la sección previa, introduzca:

```
# cdrecord padsize=63 -pad -dao -v -eject backup.iso
```

Evite hacer otras tareas con su equipo mientras se está grabando un CD o un DVD. Algunas unidades modernas tienen una funcionalidad especial que ayuda a resolver problemas relativos a que el ordenador no está respondiendo todo lo rápido que debiera. Los discos grabados con este método pueden no ser compatibles con algunos dispositivos más viejos. Si ve que las grabaciones algunas veces fallan, reduzca la velocidad. Puede cambiar la velocidad incluyendo la opción speed=, que está documentada en las páginas manual de cdrecord. Rebajar la velocidad puede ser muy importante si la imagen que se está grabando está en un sistema de archivos en red.

El relleno es necesario para que algunos lectores de CD IDE ATAPI funcionen correctamente con operaciones de lectura de cabezales que Linux y otros sistemas usan. Puede oír que las unidades más modernas funcionan sin relleno, pero

debido a que a veces, durante la lectura ocurren problemas, debería incluir está opción para asegurarse de que las unidades más antiguas también pueden leer sus grabaciones. En caso contrario, verá cómo sus archivos de copia de seguridad pudieran no funcionar en un hipotético ordenador de reemplazo.

Verificando el grabado

Después de que haya grabado un CD o DVD, es una buena idea verificar que la grabación puede leerse correctamente. El disco puede estar defectuoso, o el ordenador puede haber sufrido un golpe durante la grabación, provocando que el láser se salga de la pista de grabado.

La forma correcta de verificar una grabación es comparar todos los sectores grabados y los sectores originales del disco duro o generar sumas de comprobación de estos sectores y comprobarlas: Ambos métodos deberían usarse sólo con los sectores de datos actuales, no los sectores de relleno. El siguiente Shell script hace esta verificación bastante fácil cuando el archivo de imagen ISO original está disponible:

```
#!/bin/bash
if [[ $# -lt 1 ]] ; then
   echo "usage: isomd5 <file_or_device> ..." 1>&2
   exit 1
fi
for name in "$@" ; do
   isoinfo -di "${name}" 1>/dev/null || exit 1
done
for name in "$@" ; do
   count=( $( isoinfo -di "${name}" \
      | egrep "^Volume size is: " ) )
   count="${count[3]}"
   bsize=( $( isoinfo -di "${name}" \
      | egrep "^Logical block size is: " ) )
   bsize="${bsize[4]}"
   md5=$( dd \
      if="${name}" \
      ibs="${bsize}" \
      obs=4096 count="${size}" \
      2>/tmp/isomd5.$$.err \
      | md5sum )
   if [[ $? != 0 ]] ; then
      cat /tmp/isomd5.$$.err
      rm -f /tmp/isomd5.$$.err
      exit 1
   fi
   rm -f /tmp/isomd5.$$.err
   echo "${md5:0:32}" "" "${name}"
done
```

Este script obtiene el número de sectores usado por el sistema de archivos ISO en el archivo de imagen. Limita el número de sectores leídos por el programa de suma de comprobaciones MD5 al número de sectores usados exactamente. Esto evita leer sectores de relleno, que podrían variar en número.

Hemos llamado isomd5 a este script. Puede darle el nombre del archivo de imagen ISO, así como el nombre del dispositivo de CD que normalmente usa para leer CD-R (con el nuevo CD-R reinsertado). Debería obtener un resultado similar a este:

```
amy@desk12:~$ isomd5 backup.iso /dev/sr0
d41d8cd98f00b204e9800998ecf8427e backup.iso
d41d8cd98f00b204e9800998ecf8427e /dev/sr0
amy@desk12:~$
```

La suma de comprobación del programa MD5 es la parte de 32 caracteres hexadecimales. *Si no es la misma para el archivo de imagen ISO y para el contenido de la unidad de CD-R, la grabación es defectuosa.*

Una grabación fallida suele llamarse un *coaster* que es un término inglés que hace referencia a las marcas en forma de anillo que dejan las bebidas sobre una mesa.

Cuando un disco falle, intente:

1. Repetir la grabación sobre otro disco virgen.
2. Grabar a una menor velocidad.
3. Usar diferentes marcas de discos vírgenes.

Si los fallos persisten, probablemente tendrá una unidad de grabación que es defectuosa.

Copias de seguridad en DVD

Los pasos que se muestran en esta sección son específicos de los discos CD, pero los discos DVD también pueden grabarse de manera análoga, usando el mismo software proporcionado por cdrecord o por el paquete cdrtools. Algunos DVD, sobre todo los DVD-RAM, pueden funcionar como discos duros, pero requieren una unidad especial que soporte este modo de operación.

Haciendo copias de seguridad y guardándolas en una cinta con Amanda

Las Cintas todavía son un medio para hacer copias de seguridad bastante popular. El *Advanced Maryland Automated Network Disk Archiver* (Amanda) es un paquete de código abierto que gestiona las copias de seguridad en cinta. Desarrollado en la Universidad de Maryland, está incluido en muchas distribuciones de Linux, incluyendo Debian. Las funciones de Amanda son:

- El uso de formatos de copia de seguridad tradicionales en Unix como tar y dump.
- Operaciones sobre una LAN, haciendo copias de seguridad de datos de un cliente con un servidor de cinta central.
- Soporte para replicar clientes Windows vía Archivos Compartidos.
- Soporte para dispositivos de cinta estándar, tocadiscos y apiladoras.
- Posibilidad de balancear copias de seguridad enteras en un ciclo de varios días.
- Soporte para copias incrementales con escritura diaria de cambios.
- Compresión de datos tanto en el cliente como en el servidor, o vía dispositivos que soportan compresión hardware.
- Prevención de sobreescritura accidental en el medio equivocado.
- Una estrategia de escritura en grupos de discos que permite la escritura retardada o por etapas.
- Autentificación a través de Kerberos o con su propio esquema de autentificación.
- Encriptación de datos para la protección en redes inseguras.

Instalando Amanda

Amanda tiene componentes de cliente y componentes de servidor. El cliente se usa en sistemas cuyos datos necesitan replicarse. El servidor se usa en los sistemas que realizan la copia de seguridad y escriben datos en la cinta.

Ejecute el siguiente comando para instalar Amanda en el servidor de copias de seguridad:

```
# apt-get install amanda-server
```

Ejecute el siguiente comando para instalar Amanda en cada máquina Linux cliente:

```
# apt-get install amanda-client
```

Cuando instala estos paquetes, los otros paquetes que se necesitan se incluirán automáticamente. Si desea usar el programa amplot en Amanda, tendrá que instalar el paquete gnuplot.

Amanda usa archivos en muchos directorios diferentes. Estas opciones son configurables, pero por defecto son:

- `/etc/amanda`: Archivos de configuración (servidor).
- `/root`: El archivo `/root/.amandahosts`.
- `/usr/man/man8`: Las páginas manual.
- `/usr/share/doc/amanda-common`: Archivos de documentación.
- `/usr/share/doc/amanda-client`: Archivos de documentación específicos del cliente.
- `/usr/lib`: Librerías compartidas usada por programas Amanda.
- `/usr/lib/amanda`: Programas demonio y utilidades internas.
- `/usr/sbin`: Programas de comandos.
- `/var/lib/amanda`: Estado de ejecución, log y otros archivos.

Configurando Amanda

El archivo `/etc/services` debería tener entradas con los siguientes nombres y números de puerto. Si estas entradas no están presentes, edite el archivo `/etc/services` y luego añádalas al final. Los comentarios son opcionales:

```
/etc/services:
amanda      10080/udp   # amanda backup services
amandaidx   10082/tcp   # amanda backup services
amidxtape   10083/tcp   # amanda backup services
```

También necesitará editar el archivo `/etc/inetd.conf`, que debería contener las siguientes entradas:

```
/etc/inetd.conf: (para clientes)
amanda    dgram  udp wait    backup /usr/sbin/tcpd /usr/lib/amanda/amandad

/etc/inetd.conf: (para el servidor)
amandaidx stream tcp nowait backup /usr/sbin/tcpd /usr/lib/amanda/amindexd
amidxtape stream tcp nowait backup /usr/sbin/tcpd /usr/lib/amanda/amidxtaped
```

La primera entrada, llamada Amanda, se necesita en todos los clientes. Las otras dos entradas se necesitan sólo en el servidor. Si estas líneas no están presentes, edite el archivo `/etc/inetd.conf` y añádalas al final.

Amanda usa puertos aleatorios después de la comunicación inicial. Debería usar Amanda en Internet sólo a través de una VPN. Esto evita la necesidad de abrir un amplio rango de puertos de Internet en su LAN.

Amanda se ejecuta como el usuario de copia de seguridad con permisos del grupo disco. Necesitará establecer permisos para todos los archivos que quiera replicar para que puedan ser leídos por Amanda.

El servidor Amanda necesita estar bien conectado a la red local, con suficiente ancho de banda para el volumen de todos los datos que van a transferirse. Debería tener un disco lo suficientemente grande como para contener el espacio suficiente para poder almacenar dos veces el tamaño de la memoria a duplicar. También es necesario una CPU rápida por si el servidor va a realizar compresión hardware. Amanda soporta múltiples y diferentes configuraciones. Cada configuración consiste en un conjunto de tres archivos en un subdirectorio de `/etc/amanda`:

- `amanda.conf`: El archivo de configuración principal. Edítelo para especificar la lista de discos (vea el siguiente elemento), el dispositivo de cinta, la frecuencia de copiado, su dirección de correo, el formato de los informes y un gran vector de otras opciones.
- `disklist`: Este archivo especifica los equipos y los discos que se replicarán.
- `tapelist`: Este archivo lista las cintas activas, incluyendo las fechas en las que se escribió. Amanda gestiona este archivo, por lo que podrá verlo, pero no debería editarlo.

Nota: Mostrar todos los detalles acerca de las opciones de Amanda nos llevaría muchas páginas, por lo que dejaremos la exploración en sus manos. Los archivos de ejemplo pueden ser muy útiles debido a la gran cantidad de comentarios que ofrecen, podrá encontrarlo en el directorio `/etc/amanda/DailySet1` cuando instala el paquete Debian amanda-server. Para detalles de estos archivos de configuración, vea las páginas manual de Amanda o visite `http://wiki.zmanda.com`.

Amanda genera un informe por cada copia de seguridad que efectúa. Estos informes detallados se mandan por correo electrónico al usuario especificado en la opción mailto del archivo de configuración `amanda.conf`. Debería revisar los informes regularmente, sobre todo para comprobar errores y revisar las ejecuciones.

Restaurando archivos replicados con Amanda

Amanda usa el formato estándar de Unix para copias de seguridad (tar o dump), que debería especificar en el archivo de configuración. Esto permite usar las cintas con las copias de seguridad para restaurar los archivos incluso si el sistema

Amanda no está presente. Esto puede ser crucial a la hora de restaurar archivos después de que se estropee un disco por completo.

Amanda también ofrece herramientas de reconversión indizada que permiten la restauración de los archivos seleccionados. Asegúrese de configurar el índice para hacer que Amanda cree los archivos de índice. Las páginas manual de amrecover le ofrecerán todos los detalles.

Replicando datos MySQL

Hasta ahora, hemos estado replicando archivos y directorios. Las bases de datos tienen aspectos especiales que deben tener en cuenta. Nuestros ejemplos usan MySQL, pero los mismos principios pueden usarse con PostgreSQL y otras bases de datos relacionales.

Si nuestro servidor MySQL no necesita estar disponible 24 horas al día, 7 horas a la semana, una forma muy fácil y rápida para replicar los datos es:

1. Pare el servidor MySQL:

```
# /etc/init.d/mysqld stop
```

2. Copie los archivos de datos y los directorios de MySQL. Por ejemplo, si su directorio MySQL de datos es `/var/lib/mysql` y quiere copiarlo a `/tmp/mysql-backup`:

```
# cp -r /var/lib/mysql /tmp/mysql-backup
```

   En lugar de cp, puede usar rsync, tar, gzip u otros comandos mencionados anteriormente en este capítulo.

3. Inicie el servidor de nuevo:

```
# /etc/init.d/mysqld start
```

Las copias en línea se prestan a más trucos. Si tiene dos tablas MyISAM mutuamente independientes (sin claves ajenas o transacciones), podría bloquearlas, copiar todos sus archivos y desbloquearlas. Pero podría tener tablas InnoDB, o alguien podría ejecutar una transacción que involucre varias tablas. Afortunadamente hay varias soluciones no comerciales bastante razonables, entre las que se incluyen mysqlhotcopy, mysqlsnapshot, replication y mysqldump. Mysqlhotcopy es un script de Perl que hace copias de seguridad en línea para tablas ISAM o MyISAM.

Las páginas manual incluyen muchas opciones, pero vamos a mostrar cómo realizar la copia de seguridad de una base de datos sencilla llamada drupal:


```
# mysqlhotcopy -u user -p password drupal /tmp
Locked 57 tables in 0 seconds.
Flushed tables ('drupal'.'access', 'drupal'.'accesslog', 'drupal'.
'aggregator_
category', 'drupal'.'aggregator_category_feed', 'drupal'.'aggregator_
category_item',
'drupal'.'aggregator_feed', 'drupal'.'aggregator_item', 'drupal'.
'authmap', 'drupal'.
'blocks', 'drupal'.'book', 'drupal'.'boxes', 'drupal'.'cache',
'drupal'.'client',
'drupal'.'client_system', 'drupal'.'comments', 'drupal'.'contact',
'drupal'.'file_
revisions', 'drupal'.'files', 'drupal'.'filter_formats', 'drupal'.
'filters',
'drupal'.'flood', 'drupal'.'forum', 'drupal'.'history', 'drupal'.
'locales_meta',
'drupal'.'locales_source', 'drupal'.'locales_target', 'drupal'.'menu',
'drupal'.
'node', 'drupal'.'node_access', 'drupal'.'node_comment_statistics',
'drupal'.'node_
counter', 'drupal'.'node_revisions', 'drupal'.'permission', 'drupal'.
'poll',
'drupal'.'poll_choices', 'drupal'.'poll_votes', 'drupal'.'profile_fields',
'drupal'.
'profile_values', 'drupal'.'role', 'drupal'.'search_dataset', 'drupal'.
'search_
index', 'drupal'.'search_total', 'drupal'.'sequences', 'drupal'.
'sessions', 'drupal'.
'system', 'drupal'.'term_data', 'drupal'.'term_hierarchy', 'drupal'.
'term_node',
'drupal'.'term_relation', 'drupal'.'term_synonym', 'drupal'.'url_alias',
'drupal'.
'users', 'drupal'.'users_roles', 'drupal'.'variable', 'drupal'. 'vocabulary',
'drupal'.'vocabulary_node_types', 'drupal'.'watchdog') in 0 seconds.
Copying 171 files...
Copying indices for 0 files...
Unlocked tables.
```

mysqlhotcopy ha copiado 57 tablas (171 archivos) en 1 segundo. Mysqlsnapshot es incluso más fácil. Copia todas las tablas ISAM o MyISAM en su servidor con un archivo tar por base de datos:

```
# ./mysqlsnapshot -u user -p password -s /tmp --split -n
checking for binary logging... ok
backing up db drupal... done
backing up db mysql... done
backing up db test... done
snapshot completed in /tmp
```

Verá `mysqlsnapshot` en `http://jeremy.zawodny.com/mysql/mysqlsnapshot`.

Si ha configurado la replicación MySQL para que esté disponible 24 horas al día los 7 días de la semana, puede replicar los datos desde un servidor esclavo usando alguno de los métodos descritos. También necesitará guardar información de replicación (logs, archivos de configuración, etc). Vea los capítulos anteriores para más detalles.

Para una protección extra contra fallos en el hardware (pero no contra errores humanos), configure la replicación y configure un esclavo (y/o maestro) como un RAID 1 (en espejo).

Muchos sitios MySQL migran muchos datos desde tablas MyISAM a tablas InnoDB para conseguir verdaderas transacciones y un mejor rendimiento de escritura. Los autores del módulo InnoDB tienen un producto comercial para las copias de seguridad en línea llamado InnoDB Hot Backup que puede pedirlo desde `http://www.innodb.com/order.php`.

El último método es el que normalmente se menciona el primero en la mayoría de las documentaciones: mysqldump. En lugar de una copia en crudo, mysqldump hace un volcado ASCII de las bases de datos y de las tablas especificadas. Funciona con todos los tipos de tabla MySQL, incluidas las InnoDB. Es relativamente lento y los archivos de texto que se generan son grandes, aunque se comprimen muy bien. Es útil crear volcados de vez en cuando, puesto que contienen un script que permite restaurar sus bases de datos y tablas desde cero. Puede usar editores, grep y otras herramientas de texto para buscar en o modificar los archivos volcados.

Para bloquear todas las tablas y volcarlas a un único archivo, introduzca:

```
# mysqldump -u user -ppassword -x --all-databases > /tmp/mysql.dump
```

Puede crear una tubería a través de gzip para ahorrar algo de tiempo y de espacio:

```
# mysqldump -u user -ppassword -x --all-databases | gzip > /tmp/
mysql.dump.gz
```

Una nueva herramienta de código abierto (descarga gratuita, pago por soporte) llamada Zmanda Recovery.

Manager para MySQL ofrece una interfaz útil para muchas de estas alternativas. El sitio Web de Zmanda (`http://www.zmanda.com/backup-mysql.html`) tiene todos los detalles, pero seguidamente mencionaremos algunas de sus distintas funcionalidades:

- Tiene una interfaz de línea de comandos.
- Replica bases de datos locales o bases de datos remotas sobre SSL.
- Envía por correo electrónico el estado del procedimiento de replicación.

- Maneja todos los tipos de tablas, incluyendo InnoDB.
- No ofrece nuevos métodos de replicación. En cambio, permite escoger entre mysqldump, mysqlhotcopy, MySQL replication o LVM snapshots.
- Soporta la restauración de una transacción determinada o de un punto en el tiempo.

Zmanda ofrece archivos .tar.gz y .rpm para muchas distribuciones Linux. Para un how-to de la instalación en Debian, visite: `http://www.howtoforge.com/mysql_zrm_debian_sarge`.

Apéndice

Bash scripts de ejemplo



Este apéndice contiene varios scripts que pueden serle útil en su trabajo diario, así como para servir de modelos para escribir otros script. Puede descargar los scripts (comentados en inglés) desde `http://www.centralsoft.org`.

Añadiendo usuarios

Si usted tiene mucha tarea (como en una universidad, donde entran nuevos estudiantes al mismo tiempo o varias veces al año), este script puede ayudarle a añadirlos al sistema de forma rápida. Lee un archivo que contiene información sobre cada usuario e invoca useradd con los parámetros adecuados (vea capítulos anteriores para más detalles sobre useradd y sus variantes):

```
#!/bin/bash

expiredate=2009-02-18

if [[ -z "$1" ]] ; then
   echo ""
   echo "Please give exactly one file name."
   echo "The file will have one user per line."
   echo "Each line will have:"
   echo "   username"
   echo "   group"
   echo "   personal real name"
   echo ""
   echo "Sample line:
   echo "alfredo marketing Alfredo de Darc"
```

```
      exit 1
   fi

   cat "$1" | while read username groupname realname
   do
      # Skip blank lines.
      if [[ -z $username || -z $groupname || -z $realname ]]; then
         continue
      fi

      # Check whether the user already exists.
      # If so, report this and skip this user.
      result=$( egrep "^$username:" < /etc/passwd )
      if [[ -n "$result" ]] ; then
         echo "User '$username' already exists"
         continue
      fi

      # Check whether the group already exists.
      # If not, add the group.
      result=$( egrep "^$groupname:" < /etc/group )
      if [[ -z "$result" ]] ; then
         groupadd "$groupname"
      fi
      # Add the user.
      useradd -c "$realname" \
         -d "/home/$username" \
         -e "$expiredate"     \
         -f 365               \
         -g "$groupname"      \
         -m                   \
         -s /bin/bash         \
         "$username"

      if [[ $? == 0 ]]; then
         echo "Successfully added user '$username'."
      else
         echo "Error adding user '$username' (group \
         '$groupname', real name '$realname')"
         exit 1
      fi
   done
```

Generador de contraseñas aleatorias

Aquí mostramos un script que genera una contraseña de una longitud dada, en caracteres ASCII:

```
#!/bin/bash
n="$1"
```



```
[[ -n "$n" ]] || n=12
if [[ $n -lt 8 ]]; then
   echo "A password of length $n would be too weak"
   exit 1
fi
p=$( dd if=/dev/urandom bs=512 count=1 2>/dev/null \

   | tr -cd 'a-zA-Z0-9' \
   | cut -c 1-$n )

echo "${p}"
```

Si es capaz de mejorar la funcionalidad, merece una recompensa. Mientras que usted está fuera, nosotros seremos un poco más puntillosos con los fallos de este código. Este código es el típico que suele heredarse de un desarrollador anterior: sin comentarios, con nombres de variables poco descriptivos y algunos conjuros mágicos. Si quiere hacer del mundo un lugar mejor, hay unas cuantas cosas que puede hacer cuando escriba scripts como este.

Por último, puede escribir comentarios describiendo el propósito del código. Estos comentarios deben dividirse en dos partes: una gran cabecera de propósito general (por ejemplo, indicando los parámetros que se le pasan al script, y los valores que se tomarían por defecto), y explicaciones explícitas para comprender la dificultad de los procesos.

No pierda tiempo ejecutando los comandos básicos usados, debido a que el que lo mantenga puede mirarlos si no está familiarizado con ellos. No obstante, donde emplee una variante exótica de un comando, debería describir explícitamente su efecto y cómo puede ejecutarlo.

Después de todo, el objetivo es que documente los resultados que persigue con los conjuntos de comandos y por qué se obtienen estos resultados de la forma que ha ideado.

Ahora, vamos a dar una explicación al código del generador de contraseñas en detalle, al contrario de lo que encontrará en el mundo real. El script comienza con el comentario de inicio usual que le indica al sistema que ejecuta el intérprete bash. Luego, asignamos el primer parámetro a la variable n, que será el número de caracteres a generar. Lo ponemos entre comillas porque puede ser una cadena nula si el script se ejecuta sin parámetros. Esta cadena se prueba para determinar si es una cadena nula. El parámetro –n significa "longitud distinta de cero" por lo que si la comprobación es verdadera se ha proporcionado una cadena.

Las dos barras verticales ejecutarán la asignación que continúa si la comprobación falla. Esto obliga a que la longitud por defecto de nuestra contraseña sea de 12. Las siguientes líneas comprueban si la longitud dada es muy pequeña; hemos decidido (basándonos en las recomendaciones clásicas de los expertos en seguridad) que la longitud mínima debería ser 8.

La primera sentencia del bucle usa tres comandos del sistema en una tubería para generar una contraseña de prueba. Las tres líneas de la tubería están en un $() para capturar la salida como una cadena que luego se asignará a la variable p.

Para generar una contraseña aleatoria, necesitamos una fuente de datos aleatorios; el sistema ofrece esto combinando una variedad de fuentes estadísticas en el pseudodispositivo /dev/urandom. El comando dd lee datos binarios del dispositivo. El comando tr con la opción -cd borra todos los caracteres que no están en los rangos a-z, A-Z y 0-9. El último comando de tubería, cut, extrae el número deseado de caracteres.

Nota: No intente ejecutar este comando en su terminal y ver los resultados en la pantalla. Se quedará ciego 10 minutos y su perro empezará a maullar. ¿Todavía tiene la tentación? Deberá ejecutar un comando normal en la stty para restaurar la pantalla a un estado útil.

Búsqueda del DNS autoritativo

El script usa el comando dig que se describió en el al comienzo para búsquedas DNS, consultando la caché del servidor DNS caché local. Una funcionalidad de este script es que usa su propio nombre para especificar el tipo de registro DNS que busca. Si el script se llama a, busca registros A. Si se llama soa, busca registros DNS SOA. El nombre ptr es un caso especial que coge una dirección IPv4 y la convierte en la forma adecuada in-addr.arpa para hacer la búsqueda actual. Debería hacer una copia de este script con el nombre apropiado para cada uno de los tipos de registro comunes en DNS que podría necesitar buscar; a, aaaa, mx, etc. También puede usar enlaces duros o enlaces simbólicos para crear los alias.

Independientemente del nombre, el script acepta una lista de equipos a buscar como parámetro:

```
#!/bin/bash
#-----------------------------------------------------------------
# Copyright &#169; 2006 - Philip Howard - All rights reserved
#
# script a, aaaa, cname, mx, ns, ptr, soa, txt
#
# purpose Perform direct DNS lookups for authoritative DNS
# data. This lookup bypasses the local DNS cache
# server.
#
# syntax a      [ names ... ]
#        aaaa   [ names ... ]
#        any    [ names ... ]
#        cname  [ names ... ]
```



```
#        mx     [ names ... ]
#        ns     [ names ... ]
#        ptr    [ names ... ]
#        soa    [ names ... ]
#        txt    [ names ... ]
#
# author Philip Howard
#-----------------------------------------------------------------

# For use with ptr query.
function inaddr {
   awk -F. '{print $4 "." $3 "." $2 "." $1 ".in-addr.arpa.";}'
}

query_type=$( exec basename "${0}" )
# Get and query for each host.
for hostname in "$@" ; do
   if [[ "${query_type}" == ptr ]] ; then
   # A typical scripting trick: when a case can begin
   # with a numeral, place a dummy character such as x in
   # front because the case syntax expects an alphanumeric
   # character.
   case "x${hostname}y" in
      ( x[0-9]*\.[0-9]*\.[0-9]*\.[0-9]*y )
      hostname=$( echo "${hostname}" | inaddr )
      ;;
      ( * )
      ;;
   esac
   fi

   # Execute the query.
   dig +trace +noall +answer "${query_type}" "${hostname}" | \
      egrep "^${hostname}"
done
exit
```

Enviando archivos entre sesiones shell

Puede usar el script presentado en esta sección para enviar un archivo, o un directorio de archivos (incluyendo todos los subdirectorios), desde un sistema a otro usando una sesión shell en cada sistema. El script funciona creando un demonio rsync (rsync se describió en el último capítulo) de fondo para enviar el archivo específico o directorio. Muestra varias formas distintas de cómo puede usarse para recibir un archivo o directorio. Este script no necesita estar en el sistema receptor, por lo que puede usarse para enviarse una copia de sí mismo. El paquete rsync, sin embargo, debe instalarse en ambos sistemas.

Nota: El sistema remitente debe tener acceso a la red por el número de puerto que se usa para aceptar las conexiones rsync entrantes. El número de puerto se escoge de manera aleatoria entre el rango 12288 y 28671. Puede sobrescribir la elección aleatoria de puerto usando la opción –p seguida de un número de puerto. Si las reglas de su cortafuegos sólo permiten unos cuantos puertos a los que conectarse, debe elegir uno de estos puertos en este script.

Para transferir datos, primero ejecute el script en el sistema remitente. Una vez que se muestren los comandos de ejemplos, seleccione qué comando es el apropiado para usarse basándose en la dirección IP o el nombre del equipo que puede alcanzar el sistema remitente, y la ubicación donde el archivo o directorio se almacenará en el sistema receptor. Copie la línea de comando seleccionada y pegue el comando en la Shell del sistema receptor para ejecutar el comando rsync que recibe los datos. El demonio se continuará ejecutando cuando la transferencia esté completa, permitiéndole transferir un archivo o directorios muchas veces entre computadores diferentes. Pare el demonio cuando las transferencias estén completas presionando **Control-C** en la ventana de la shell del sistema remitente.

Nota: Este script no tiene seguridad. Cualquiera que sepa la dirección y el número de puerto en el que se está escuchando, puede obtener los datos que se están transfiriendo. No debería usar este script para transferir datos secretos o confidenciales, intente scp o sftp en su lugar. Asegúrese de parar el demonio una vez que las transferencias deseadas estén completadas.

El nombre que se sugiere para este script es rsend:

```
#!/bin/bash
#---------------------------------------------------------------
# Copyright &#169; 2006 - Philip Howard - All rights reserved
#
# script rsend
#
# purpose   To start an rsync daemon in the shell foreground
#           to send a specified directory or file when
#           retrieved using one of the rsync command lines
#           shown, by pasting it in a shell session on another
#           host.
#
# usage     rsend    [options]   directory   |  file
#
# options   -c include checksum in the rsync command lines
#           -d change daemon to the specified directory
#           -n include dryrun in the rsync command lines
#           -p use the specified port number, else random
```



```
#           -s include sparse in the rsync command lines
#           -u user to run as, if started as root
#           -v show extra information
#
# author Philip Howard
#---------------------------------------------------------------
umask 022
hostname=$( exec hostname -f )
whoami=$( exec whoami )
uid="${whoami}"

#---------------------------------------------------------------
# Set defaults.
#---------------------------------------------------------------
checksum=""
delete=""
delmsg=""
dryrun=""
padding="------"
port=""
sparse=""
verbose=""

bar1="-------------------------"
bar1="#${bar1}${bar1}${bar1}"

bar2="#########################"
bar2="#${bar2}${bar2}${bar2}"

#---------------------------------------------------------------
# Include paths for ifconfig.
#---------------------------------------------------------------
export PATH="${PATH}:/usr/sbin:/sbin"

#---------------------------------------------------------------
# Scan options.
#---------------------------------------------------------------
while [[ $# -gt 0 && "x${1:0:1}" = "x-" ]]; do
   case "x${1}" in
   ( x-c | x--checksum )
   checksum="c"
   ;;
   ( x--delete )
   delete=" --delete"
   delmsg="/delete"
   padding=""
   ;;
   ( x-d | x--directory )
   shift
   cd "${1}" || exit 1
   ;;
   ( x--directory=* )
```

```
      cd "${1:12}" || exit 1
      ;;
      ( x-n | x--dry-run )
      dryrun="n"
      ;;
      ( x-p | x--port )
      shift
      port="${1}"
      ;;
      ( x--port=* )
      port="${1:7}"
      ;;
      ( x-s | x--sparse )
      sparse="S"
      ;;
      ( x-u | x--user )
      shift
      uid="${1}"
      ;;
      ( x--user=* )
      uid="${1:7}"
      ;;
      ( x-v | x--verbose )
      verbose=1
      ;;
      esac
      shift
done

#-----------------------------------------------------------------
# Get a random number for a port.
#-----------------------------------------------------------------
if [[ -z "${port}" || "${port}" = 0 || "${port}" = . ]]; then
   port=$( dd if=/dev/urandom ibs=2 obs=2 count=1 2>/dev/null \
      | od -An -tu2 | tr -d ' ' )
   port=$[ $port % 16384 ]
   port=$[ $port + 12288 ]
fi

#-----------------------------------------------------------------
# Make up names for temporary files to be used.
#-----------------------------------------------------------------
conffile="/tmp/rsync-${whoami}-${port}-$$.conf"
lockfile="/tmp/rsync-${whoami}-${port}-$$.lock"

#-----------------------------------------------------------------
# This function adds quotes to strings that need them.
# Add single quotes if it has one of these: space $ " `
# Add double quotes if it has one of these: '
# Note: not all combinations will work.
#-----------------------------------------------------------------
function strquote {
```



```
   local str
   str=$( echo "${1}" | tr -d ' $"`' )
   if [[ "${str}" != "${1}" ]]; then
   echo "'${1}'"
   return
   fi
   str=$( echo "${1}" | tr -d "'" )
   if [[ "${str}" != "${1}" ]]; then
   echo '"'"${1}"'"'
   return
   fi
   echo "${1}"
   return 0
}

#-----------------------------------------------------------------
# Only one name can be handled.
#-----------------------------------------------------------------
if [[ $# -gt 1 ]]; then
   echo "Only one name (directory or file)" 1>&2
   exit 1
elif [[ $# -eq 1 ]]; then
   name="${1}"
else
   name=$( exec pwd )
fi

#-----------------------------------------------------------------
# Set up a temporary config file.
#
# Arguments:
# $1 Directory transferred, or where transfer is starting
# $2 Not used (AO: Should be removed)
# $3 File transferred (if single file specified)
#-----------------------------------------------------------------
function configout {
   echo "lock file = ${lockfile}"
   echo "log file = /dev/stderr"
   echo "use chroot = false"
   echo "max connections = 32"
   echo "socket options = SO_KEEPALIVE"
   echo "list = yes"
   echo "[.]"
   echo "path = ${1}"
   echo "read only = yes"
   echo "uid = ${uid}"
   echo "comment = ${2}"
   if [[ -n "${3}" ]]; then
   echo "include = **/${3}"
   echo "exclude = **"
   fi
```

```
}
#------------------------------------------------------------------
# Get directory and file.
#------------------------------------------------------------------
if [[ ! -e "${name}" ]]; then
   echo "does not exist:" $( strquote "${name}" ) 1>&2
   exit 1
elif [[ -d "${name}" ]]; then
   p=$( exec dirname "${name}" )
   b=$( exec basename "${name}" )
   d="${name}"
   f=""
   r=$( cd "${name}" && exec pwd )
   announce="${d}"
   rsyncopt="-a${checksum}${dryrun}H${sparse}vz${delete}"
   configout "${d}/." "directory:${d}/" >"${conffile}"
elif [[ -f "${name}" ]]; then
   p=$( exec dirname "${name}" )
   b=$( exec basename "${name}" )
   d="${p}"
   f="${b}"
   r=$( cd "${p}" && exec pwd )
   r="${r}/${b}"
   announce="${d}/${f}"
   rsyncopt="-a${checksum}${dryrun}${sparse}vz"
   configout "${d}/." "file:${d}/${f}" >"${conffile}"
elif [[ -L "${name}" ]]; then
   p=$( exec dirname "${name}" )
   b=$( exec basename "${name}" )
   d="${p}"
   f="${b}"
   r=$( cd "${p}" && exec pwd )
   r="${r}/${b}"
   announce="${d}/${f}"
   rsyncopt="-a${checksum}v"
   configout "${d}/." "symlink:${d}/${f}" "${f}" >"${conffile}"
fi

#------------------------------------------------------------------
# Show config file if verbose is requested.
#------------------------------------------------------------------
if [[ -n "${verbose}" ]]; then
   echo "${bar2}"
   ls -ld "${conffile}"
   echo "${bar2}"
   cat "${conffile}"
fi

#------------------------------------------------------------------
# This function outputs example receive commands.
#------------------------------------------------------------------
function showrsync {
```



```
   echo -n "rsync ${rsyncopt} "
   if [[ -n "${oldfmt}" ]]; then
   echo "--port=${port}" $( strquote "${1}::${2}" ) $( strquote "${3}" )
   else
   echo $( strquote "rsync://${1}:${port}/${2}" ) $( strquote "${3}" )
   fi
   return
}

#------------------------------------------------------------------
# These functions show rsync commands for hostname and IP address.
#------------------------------------------------------------------
function getip {
   case $( exec uname -s ) in
   ( SunOS )
   netstat -i -n | awk '{print $4;}'
   ;;
   ( Linux )
   ifconfig -a | awk '{if($1=="inet")print substr($2,6);}'
   ;;
   ( * )
   netstat -i -n | awk '{print $4;}'
   ;;
   esac
   return
}

function ipaddr {
   getip \
   | egrep '^[0-9]*\.[0-9]*\.[0-9]*\.[0-9]*$' \
   | egrep -v '^0\.|^127\.' \
   | head -2 \
   | while read ipv4 more ; do
     showrsync "${ipv4}" "$@"
   done
   return
}

function showcmd {
   ipaddr "${2}" "${3}"
   showrsync "${1}" "${2}" "${3}"
   return
}

#------------------------------------------------------------------
# Announce the shell commands to receive this data.
#------------------------------------------------------------------
echo "${bar2}"
echo "# sending ${announce}"
echo "# paste ONE of these commands in a remote shell to receive"

if [[ -d "${name}" ]]; then
```

```
    echo "${bar1}"
    showcmd "${hostname}" . .

    echo "${bar1}"
    showcmd "${hostname}" . "${b}"

    if [[ "${d}" != "${b}" && "${d}" != "${r}" ]]; then
    echo "${bar1}"
    showcmd "${hostname}" . "${d}"
    fi

    echo "${bar1}"
    showcmd "${hostname}" . "${r}"
else
    echo "${bar1}"
    showcmd "${hostname}" "./${f}" "${b}"
    s=$( exec basename "${d}" )
    s="${s}/${f}"
    if [[ "${s}" != "${b}" ]]; then
    echo "${bar1}"
    showcmd "${hostname}" "./${f}" "${s}"
    fi

    if [[ "${name}" != "${b}" \
       && "${name}" != "${s}" \
       && "${name}" != "${r}" ]]; then
    echo "${bar1}"
    showcmd "${hostname}" "./${f}" "${name}"
    fi

    echo "${bar1}"
    showcmd "${hostname}" "./${f}" "${r}"
fi

echo "${bar1}"
echo "# press ^C here when done"
echo "${bar2}"

#-----------------------------------------------------------------
# Start rsync in daemon mode.
#-----------------------------------------------------------------
s="DONE"
trap 's="SIGINT ... DONE"' INT
trap 's="SIGTERM ... DONE"' TERM
rsync --daemon --no-detach "--config=${conffile}" "--port=${port}"
rm -f "${conffile}" "${lockfile}"
echo "${s}"
```

Integrando ssh y screen

Debería estar familiarizado con el comando ssh, que se conecta a otro equipo e inicia una shell allí de manera segura. El comando screen es una herramienta

útil que permite que una sesión Shell quede en un estado activo, con la pantalla intacta cuando desconecta del ordenador remoto. La sesión capturada podrá retomarse después, incluso desde un equipo diferente. También es posible tener dos o más conexiones en la misma sesión Shell.

El siguiente script hace una conexión ssh e inicia una sesión llamada screen en un comando. El beneficio de usar este script es que es más rápido para conectarse y desconectarse cuando se está trabajando con múltiples servidores.

Este script se usa de manera similar al comando ssh. La sintaxis ssh que especifica el nombre de usuario y el equipo de la sesión remota se expande para incluir el nombre de sesión. Puede crear múltiples sesiones en un equipo remoto con el mismo nombre de usuario y diferentes sesiones. El nombre de sesión es opcional. Si no se proporciona, el script ejecuta el comando ssh de una manera normal, sin ejecutar screen. La sintaxis completa de este script, incluyendo las opciones ssh que soporta, pueden verse en los comentarios del script.

El nombre sugerido para este script es ss:

```
#!/usr/bin/env bash
#-----------------------------------------------------------------
# Copyright &#169; 2006 - Philip Howard - All rights reserved
#

# command ss (secure screen)
#
# purpose Establish a screen based background shell session
# via secure shell communications.
#
# syntax    ss  [options] session/username@hostname
#           ss  [options] session@username@hostname
#           ss  [options] username@hostname/session
#           ss  [options] username@hostname session
#
# options   -h hostname
#           -h=hostname
#           -i identity
#           -i=identity
#           -l loginuser
#           -l=loginuser
#           -m Multi-display mode
#           -p portnum
#           -p=portnum
#           -s session
#           -s=session
#           -t Use tty allocation (default)
#           -T Do NOT use tty allocation
#           -4 Use IPv4 (default)
#           -6 Use IPv6
#           -46 | -64 Use either IPv6 or IPv4
#
```

```
# requirements The local system must have the OpenSSH package
#            installed. The remote system must have the
#            OpenSSH package installed and have the sshd
#            daemon running. It must also have the screen(1)
#            program installed. Configuring a .screenrc
#            file on each system is recommended.
#
# note       The environment variable SESSION_NAME will be set
#            in the session created under the screen command
#            for potential use by other scripts.
#
# author Philip Howard
#-----------------------------------------------------------------
whoami=$( exec whoami )
hostname=$( exec hostname )

h=""
i=( )
m=""
p=( )
s=''
t=( -t )
u="${whoami}"
v=( -4 )

#-----------------------------------------------------------------
# Parse options and arguments.
#-----------------------------------------------------------------
while [[ $# -gt 0 ]]; do
   case "x${1}" in
   ( x*/*@* )
   # Example: session1/lisa@centrhub
   u=$( echo "x${1}" | cut -d @ -f 1 )
   u="${u:1}"
   s=$( echo "x${u}" | cut -d / -f 2 )
   u=$( echo "x${u}" | cut -d / -f 1 )
   u="${u:1}"
   h=$( echo "x${1}" | cut -d @ -f 2 )
   shift
   break
   ;;
   ( x*@*/* )
   # Example: lisa@centrhub/session1
   u=$( echo "x${1}" | cut -d @ -f 1 )
   u="${u:1}"
   h=$( echo "x${1}" | cut -d @ -f 2 )
   s=$( echo "x${h}" | cut -d / -f 2 )
   h=$( echo "x${h}" | cut -d / -f 1 )
   h="${h:1}"
   shift
   break
   ;;
```



```
   ( x*@*@* )
   # Example: session1@lisa@centrhub
   s=$( echo "x${1}" | cut -d @ -f 1 )
   s="${s:1}"
   u=$( echo "x${1}" | cut -d @ -f 2 )
   h=$( echo "x${1}" | cut -d @ -f 3 )
   shift
   break
   ;;
   ( x*@* )
   # Example: lisa@centrhub
   u=$( echo "x${1}" | cut -d @ -f 1 )
   u="${u:1}"
   h=$( echo "x${1}" | cut -d @ -f 2 )
   # Next argument should be session name.
   shift
   if [[ $# -gt 0 ]]; then
      s="${1}"
      shift
   fi
   break
   ;;
   ( x-h=* )
   h="${1:3}"
   ;;
   ( x-h )
   shift
   h="${1}"
   ;;
   ( x-i=* )
   i="${1:3}"
   if [[ -z "${i}" ]]; then
      i=( )
   else
      i=( -i "${1:3}" )
   fi
   ;;
   ( x-i )
   shift
   i=( -i "${1}" )
   ;;
   ( x-l=* | x-u=* )
   u="${1:3}"
   ;;
   ( x-l | x-u )
   shift
   u="${1}"
   ;;
   ( x-m | x--multi )
   m=1
   ;;
   ( x-p=* )
```

```
      p="${1:3}"
      if [[ -z "${p}" ]]; then
         p=( )
      else
         p=( -p "${1:3}" )
      fi
      ;;
      ( x-p )
      shift
      p=( -p "${1}" )
      ;;
      ( x-s=* )
      s="${1:3}"
      ;;
      ( x-s )
      shift
      s="${1}"
      ;;
      ( x-t )
      t=( -t )
      ;;
      ( x-T )
      t=( )
      ;;
      ( x-4 )
      v=( -4 )
      ;;
      ( x-6 )
      v=( -6 )
      ;;
      ( x-46 | x-64 )
      v=( )
      ;;
      ( x-* )
      echo "Invalid option: '${1}'"
      die=1
      ;;
      ( * )
      echo "Invalid argument: '${1}'"
      die=1
      ;;
      esac
      shift
   done

   #------------------------------------------------------------------
   # Make sure essential information is present.
   #------------------------------------------------------------------
   if [[ -z "${u}" ]]; then
      echo "User name is missing"
      die=1
   fi
```



```
if [[ -z "${h}" ]]; then
   echo "Host name is missing"
   die=1
fi

[[ -z "${die}" ]] || exit 1

#------------------------------------------------------------------
# Run screen on the remote only if a session name is given.
#------------------------------------------------------------------
c=( ssh "${v[@]}" "${i[@]}" "${p[@]}" "${t[@]}" "${u}@${h}" )
if [[ -n "${s}" ]]; then
   o="-DR"
   [[ -n "${m}" ]] && o="-x"
   x="exec /usr/bin/env SESSION_NAME='${s}' screen ${o} '${s}'"
   c=( "${c[@]}" "${x}" )
fi
exec "${c[@]}"
```

Índice alfabético

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

